DR. SCHETTER BMC PUCHHEIM-MÜNCHEN

# MC - 12 SYSTEM

Bedienungshandbuch

**Complete Table of Contents**

## Contents

# MC - 12 SYSTEM
# Operating manual

**Important note:**

The operating program of the MC-12(A) with all its extensions has been created and tested with the greatest possible care. BMC and RVS make no warranties, either with respect to this manual or with respect to the hardware and software described in this book, its quality, feasibility or fitness for a particular purpose. BMC and RVS shall in no event be liable for any damage caused or resulting directly or indirectly from improper use or any defect in the system. We reserve the right to make changes in the interest of technical progress.



DR. SCHETTER BMC GmbH
Wettersteinstr. 4
8039 Puchheim

RVS Datentechnik GmbH
Hainbuchstr. 2
8000 Munich 45

# Table of Contents

**Measurement data acquisition system MC-12 (A)**

## Introduction

The MC-12 and MC-12A measurement data acquisition systems are very universally designed so that they are suitable for a wide range of applications for which several devices are generally required.

In multimeter mode, 5 measurement signals can be recorded continuously. Each voltage value can be subjected to any number of complicated transformations (e.g. conversion of voltage to dB, voltage to RMS value, voltage to temperature, etc.). RMS value, voltage - to temperature, etc.). Pluggable input amplifiers allow adaptation to all common sensor types.

In transient recorder mode, up to 5 measured values with sampling times from 33 us to any value can also be recorded. The measured data can be stored in the user-organizable memory with variable pre- and post-history. The basic unit has a “maximum memory depth of 7168 measuring points (option approx. 16000 or 24000).

To display the measurement signals, a standard analog oscilloscope can be connected to the analog outputs of the MC-12 (A). A cursor generated by the MC-12 allows for precise evaluation of the measurement signals.

An automatic scaling of the plot command allows the documentation of measurement results on the 4-color printer/plotter CE-150.

In addition to its diverse measuring functions, the MC-12 (A) offers a convenient control and regulation system with the help of two analog outputs and a total of six switching outputs (with the MC-12 (A) an additional 24 programmable control lines).

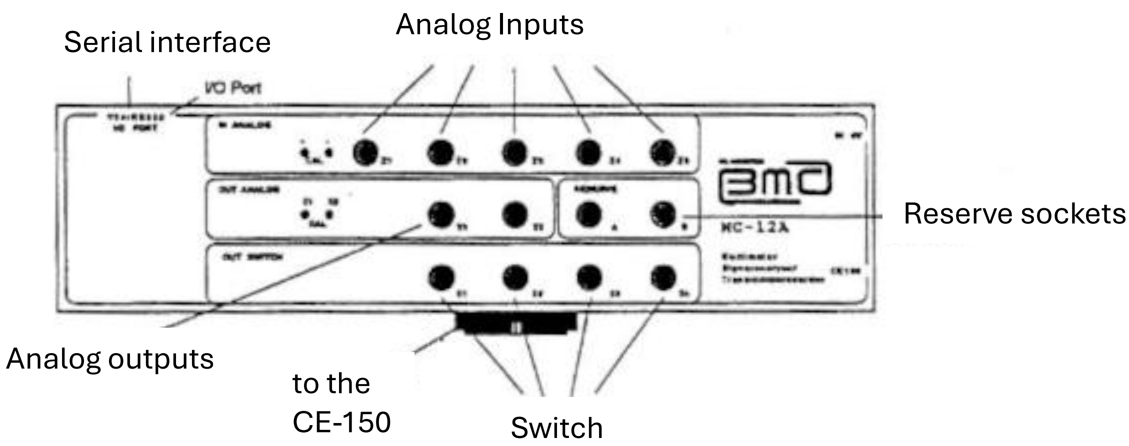
Over 40 additional BASIC commands are available to the user for specific tasks. This allows the user to solve almost all measurement, control, and regulation tasks themselves.

Many software and hardware tools are available as a supplement to the MC-12(A) system, thus significantly expanding its range of applications.

# Chapter 1

**SYSTEM DATA**

In this chapter you will find the connection values of the inputs and outputs, accuracy specifications and performance data of the MC-12(A).




## 1.1 INPUTS

The MC-12(A) system has five analog inputs, numbered 1 to 5. The measuring range can be individually adjusted for each input from 4.92 V to 0.0481 V in 11 steps. In the AUTORANGE ON mode, the measuring system always selects the most suitable measuring range.

The input resistance is 1MOhm and the bandwidth is 0 - 40 kHz. (Caution! When the MC-12(A) is turned off, the inputs are connected to ground via 1kOhm; see also Chapter 2.)

The following tables show the resolution and accuracy of the MC-12(A) in the various operating modes.

If the upper measuring range limit is exceeded, ERROR 110 is displayed. In AUTORANGE mode, this error message is only displayed for input voltages greater than 4.922 V in bipolar mode or +4.942 V in unipolar mode.

**Measuring ranges, resolution and accuracy of the MC- 12 basic system**

**Bipolar – Operation**

| No. | Measuring range/V | Resolution/V | Error/V |
|---|---|---|---|
| 1 | 4.922E+00 | 38.5 E-03 | ±38.5E-03 |
| 2 | 2.461E+00 | 19.2E-03 | ±19.2E-03 |
| 3 | 1.231E+00 | 96.1E-04 | ±96.1E-04 |
| 4 | 6.153E-01 | 48.1E-04 | ±48.1E-04 |
| 5 | 3 .076E-01 | 24.0E-04 | ±24.0E-04 |
| 6 | 1. 538E-01 | 12.0E-04 | ±12.0E-04 |
| 7 | 7.691E-02 | 60.1E-05 | ±60.1E-05 |
| 8 | 3.846E-02 | 30.0E-05 | ±30.0E-05 |
| 9 | 1.923E-02 | 15.0E-05 | ±15.0E-05 |
| 10 | 9.614E-03 | 75.1E-06 | ±15.0E-05 |
| 11 | 4.807E-03 | 37.6E-06 | ±15.0E-05 |

**Unipolar – Operation**

| No. | Measuring range/V | Resolution/V | Error/V |
|---|---|---|---|
| 1 | 4.942E+00 | 19.3E-03 | ±19.3E-03 |
| 2 | 2.471E+00 | 96.5E-04 | ±96.5E-04 |
| 3 | 1.235E+00 | 48.3E-04 | ±48.3E-04 |
| 4 | 6.177E-01 | 24.1E-04 | ±24.1E-04 |
| 5 | 3 .089E-01 | 12.1E-04 | ±12.1E-04 |
| 6 | 1. 544E-01 | 60.3E-05 | ±60.3E-05 |
| 7 | 7.721E-02 | 30.2E-05 | ±30.2E-05 |
| 8 | 3.861E-02 | 15.1E-05 | ±15.0E-05 |
| 9 | 1.930E-02 | 75.4E-06 | ±15.0E-05 |
| 10 | 9.652E-03 | 37.7E-06 | ±15.0E-05 |
| 11 | 4.826E-03 | 18.9E-06 | ±15.0E-05 |

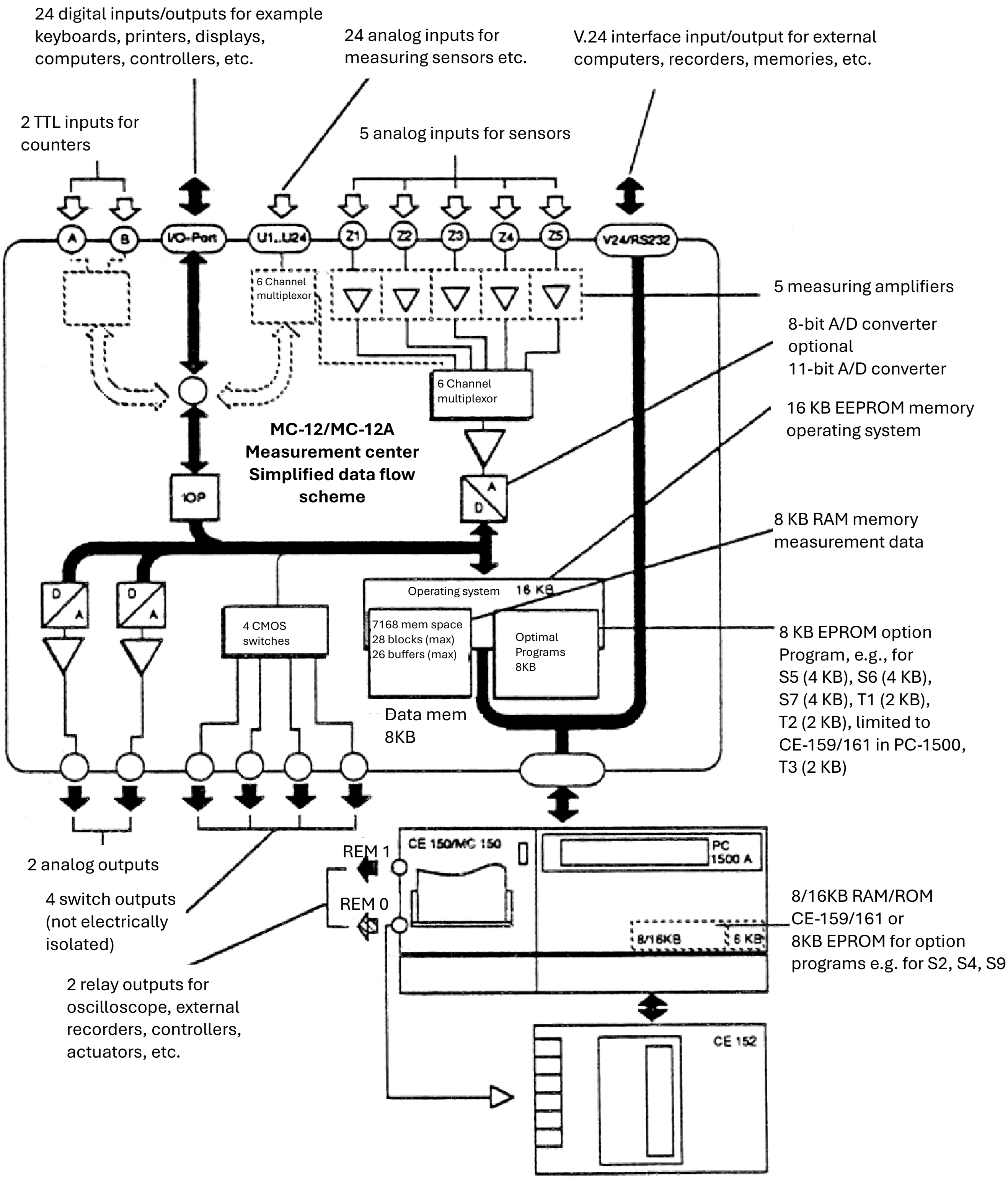
## Measuring ranges, resolution and accuracy with 11-bit module

**Bipolar – Operation**

| No. | Measuring range/V | Resolution/V | Error/V |
|---|---|---|---|
| 1 | 4.922E+00 | 4.88E-03 | ±4.88E-03 |
| 2 | 2.461E+00 | 2.44E-03 | ±2.44E-03 |
| 3 | 1.231E+00 | 1.22E-03 | ±1.22E-03 |
| 4 | 6.153E-01 | 6.10E-04 | ±6.10E-04 |
| 5 | 3 .076E-01 | 3.05E-04 | ±3.05E-04 |
| 6 | 1. 538E-01 | 1.52E-04 | ±1.52E-04 |
| 7...11 | as with 8 bit | | |

**Unipolar – Operation**

| No. | Measuring range/V | Resolution/V | Error/V |
|---|---|---|---|
| 1 | 4.942E+00 | 2.44E-03 | ±2.44E-03 |
| 2 | 2.471E+00 | 1.22E-03 | ±1.22E-03 |
| 3 | 1.235E+00 | 6.10E-04 | ±6.10E-04 |
| 4 | 6.177E-01 | 3.05E-04 | ±3.05E-04 |
| 5 | 3.089E-01 | 1.52E-04 | ±1.52E-04 |
| 6 | 1.544E-01 | 7.63E-05 | ±1.50E-04 |
| 7...11 | as with 8 bit | | |



24 digital inputs/outputs for example keyboards, printers, displays, computers, controllers, etc.
24 analog inputs for measuring sensors etc.
V.24 interface input/output for external computers, recorders, memories, etc.
2 TTL inputs for counters
5 analog inputs for sensors
A
B
I/O-Port
U1..U24
Z1
Z2
Z3
Z4
Z5
V24/RS232
6 Channel multiplexor
5 measuring amplifiers
8-bit A/D converter optional 11-bit A/D converter
6 Channel multiplexor
16 KB EEPROM memory operating system
MC-12/MC-12A Measurement center Simplified data flow scheme
IOP
A
D
8 KB RAM memory measurement data
D
A
D
A
4 CMOS switches
Operating system 16 KB
7168 mem space
28 blocks (max)
26 buffers (max)
Optimal Programs 8KB
8 KB EPROM option Program, e.g., for S5 (4 KB), S6 (4 KB), S7 (4 KB), T1 (2 KB), T2 (2 KB), limited to CE-159/161 in PC-1500, T3 (2 KB)
Data mem 8KB
2 analog outputs
4 switch outputs (not electrically isolated)
2 relay outputs for oscilloscope, external recorders, controllers, actuators, etc.
REM 1
REM 0
CE 150/MC 150
PC 1500 A
8/16KB
8 KB
8/16KB RAM/ROM CE-159/161 or 8KB EPROM for option programs e.g. for S2, S4, S9
CE 152

## 1.2 OUTPUTS

The outputs provide an analog voltage in the range of -5.0 V to +4.96 V in bipolar mode and from 0 V to +4.98 V in unipolar mode. These are not power outputs. The output current is limited to 1 mA. Monotonicity is guaranteed for the outputs. In bipolar mode, the smallest voltage step is 39 mV, in unipolar mode it is 19.5 mV. In both operating modes, 255 steps are possible (8 bits).

## 1.3 MULTIMETER

In multimeter mode, 256 measurements are taken within a period of 20ms (corresponding to one period at 50 Hz) to generate a measured value. The arithmetic mean yields the digitally displayed value. This method achieves complete mains hum suppression.

Using a special command, the measured value can be transformed with any arithmetic operation before it is displayed (see Chapter 4).

## 1.4 TRANSIENT RECORDER

With a maximum sampling frequency of 30 kHz, single events can be sampled. Up to 7000 measured values can be recorded and stored. It is possible to simultaneously record up to five input channels with different sensitivities and any history. The minimum possible sampling time depends on the desired setting and can be increased to 2000 s per measurement cycle in 50µs increments.

Output via a simple oscilloscope with cursor control allows for easy evaluation of the recorded signals. The hardcopy command (printing the oscilloscope image on a plotter) and the plot command (printing up to five signals in one printout) enable simple and clean documentation (see Chapter 5).

# Chapter 2

**OPERATING INSTRUCTIONS**

## 2.1 Power supply

The MC-12(A) system is built entirely using CMOS technology and is characterized by very low power consumption. This enables mains-independent operation for several hours. The maximum operating time depends heavily on the specific application. The following values can serve as a guide, assuming the MC-12(A) batteries are fully charged: (max. charging time: 14 hours).

4-5 hours of continuous measurement operation without turning off the MC-12(A).

6 hours of normal measurement operation, with the MC-12(A) turned off after each measurement.

The following should be noted:

The CE-150 printer's battery allows for continuous printing for a maximum of 45 minutes.

If the battery voltage of the CE-150 is no longer sufficient for printing, ERROR 78 or ERROR 80 is displayed as soon as a print command is given.

For stationary use, power is supplied by the CE-150's AC adapter. The short connecting cable, which is included with the MC-12(A), plugs one end into the CE-150's AC input jack and the other end into the MC12(A)'s AC output jack (Figure 2.1). The AC adapter is connected to the MC-12(A)'s 9V IN jack.

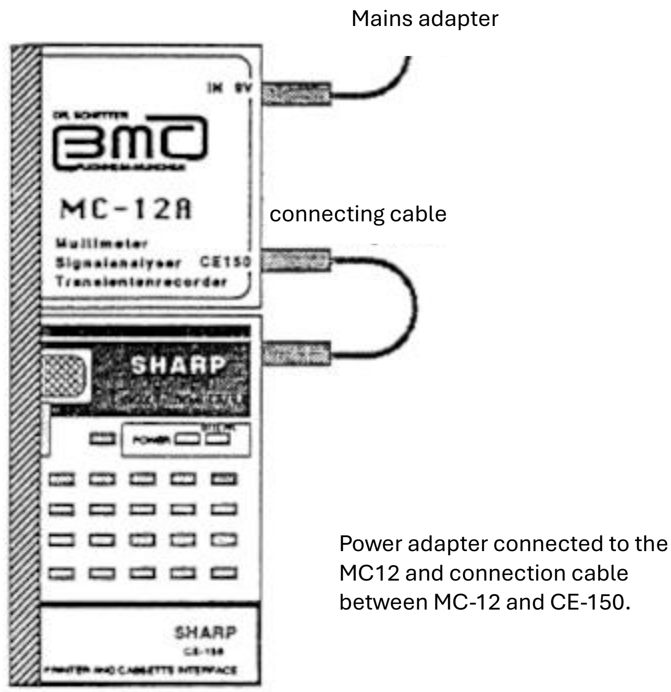



Image 2.1

If you are using an MC-150 printer (identical to the CE-150 with a different power supply), no connection cable is required. The power supply can be connected to one of the three power sockets (MC-12(A) or MC-150).

If the battery voltage is no longer sufficient for the MC-12(A), ERROR 135 is displayed as soon as the MC ON command is entered, or a measurement command is carried out while the device is already switched on (e.g.: AUCHA 1).

If ERROR 78 or ERROR 80 occurs and you want to continue working with the AC adapter, turn off the PC-1500(A) computer and connect the AC adapter as shown in Figure 2.1. Operation can be resumed after approximately 2 minutes.

## 2.2 Operating temperature

The measuring system is guaranteed to operate within a temperature range of +5°C to +40°C. If the printer cartridges are warmed up or the printer is not in use, operation down to 0°C is possible. The reduced performance of the batteries must be taken into account.

## 2.3 Connection of measuring signals

Under no circumstances should voltages higher than +-10V be connected to the input jacks. Since the MC-12(A) system is completely floating, proper grounding must be ensured in individual cases. High static voltages may cause programs to abort. Restarting the program will generally allow you to resume operation. Use the connectors supplied with the MC-12(A) system to connect your measurement signals. The jacks have the following pin assignment:

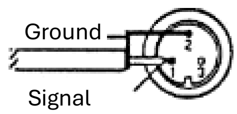



Figure 2.2: Pin assignment of the input sockets (top view)

**Be careful when connecting sensitive small signal voltages:**

As long as the MC-12(A) is turned off, the inputs are grounded via 1kOhm. This is important to note, especially if other measuring devices are connected to the same signal. When the MC-12(A) is turned on, the input resistance is 1MOhm.

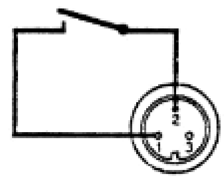
The MC-12(A) housing does not have the same potential as the signal ground. It is connected to the negative terminal of the power supply. This is especially important if both the MC-12(A) system and external sensors connected to the signal inputs are powered from a common voltage source. In this case, galvanic isolation of the MC-12(A) system from the sensors must always be ensured.

## 2.4 Switches

Four CMOS analog switches and two relays are available to control processes.

The four CMOS switches can be activated using the command ‘**`SWITCH 1..4 ON`**‘ or’ **`SWITCH 1..4 OFF`**‘

The signals to be switched must not exceed a voltage of +/-10 V, and the switching current must not exceed 80 mA. The pin assignments can be found in the connection diagram.




Switch sockets

To ensure proper operation, ensure that there is no electrical isolation between the circuit and the MC-12(A). Therefore, the circuit's ground potential must be connected to the MC-12(A)'s ground potential.

If signals are to be switched potential-free, use the two relays **REM0** and **REM1** of the CE-150.

The commands **'RELAY 0 ON'** or **'RELAY 0 OFF'** and **'RELAY 1 ON'** or **'RELAY 1 OFF'** are available to operate the relays. The **REM0** control is only effective when the **REMOTE** switch on the CE-150 is in the ON position.

## 2.5 Bipolar/unipolar switching

To enable an even wider range of applications, the MC-12(A) offers the option of operating all inputs and outputs either bipolar (-5V ... +5V) or unipolar (0V ... +5V). This doubles the resolution of voltages in the range (0V ... +5V).

For technical reasons, the measurement input (A/D converter) must be readjusted when switching operating modes. Please refer to the adjustment instructions in Appendix C.

This mode change should therefore only be performed in special cases. When the MC-12(A) is first connected to the PC-1500(A) and switched on, the bipolar mode is selected. The MC-12(A) system is delivered calibrated in bipolar mode.

## 2.6 Connecting an oscilloscope

Any two-channel oscilloscope can be used as a peripheral device to the MC-12(A) to display stored signals.

The MC-12(A)'s two analog outputs provide the signals for both channels of the oscilloscope. Connect the two oscilloscope inputs to the MC-12(A)'s analog outputs. If you are using a single-channel oscilloscope, connect only output 1.

Set the oscilloscope inputs to "DC"

The signal amplitude of the image signals is 5V, the image duration is 20ms, so that for most oscilloscopes the setting 2V/cm and 2ms/cm makes sense.

Set the trigger to "LINE." If no image appears when using the corresponding commands (see Section 11.1), adjust the trigger level until you get a steady image.

# Chapter 3

**COMMISSIONING**

The brief instructions provided here for commissioning the PC1500(A) and CE-150 cannot replace the study of the separate operating manuals. This section only briefly summarizes the most important points required for operating the MC-12(A) system.

## 3.1 Commissioning the PC-1500(A)

The PC-1500(A) is powered on by inserting the included batteries. When the PC-1500(A) is plugged onto the CE-150, the batteries in the PC-1500(A) are disabled.

Insert batteries (Caution: Do not use rechargeable NiCd batteries!)

If you want to install a memory expansion module (CE-155, CE-159, CE-161), open the module compartment on the back of the computer.

Caution: Before touching the module, you should touch a well-grounded object (wall, faucet) to prevent damage to the module due to static electricity.

Insert module (do not touch contacts)

Turn on the computer and clear the display NEW0:CHECK with CL

NEW0 <ENTER>

The computer is now ready for use. If you encounter any problems, please refer to the computer manual and repeat the setup process.

## 3.2 Commissioning of the CE-150

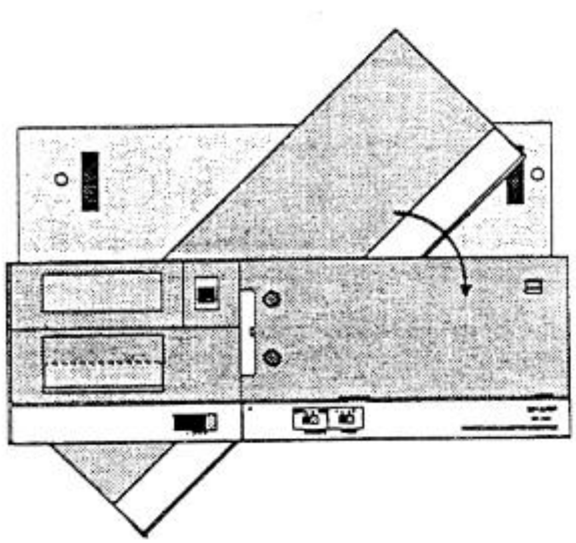
First, connect the included AC adapter to the CE-150 (socket on the bottom right side). The CE-150's NiCd batteries require approximately 5 minutes to build up sufficient voltage for operation. To achieve full battery performance, the CE-150 should be charged for at least 15 hours.

Please proceed as described in the CE-150 description.

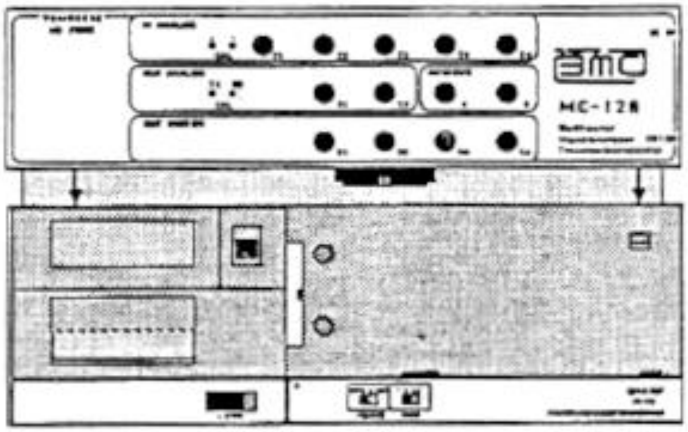
Use an angled mounting plate.

(The straight mounting plate used on older devices cannot be used.)

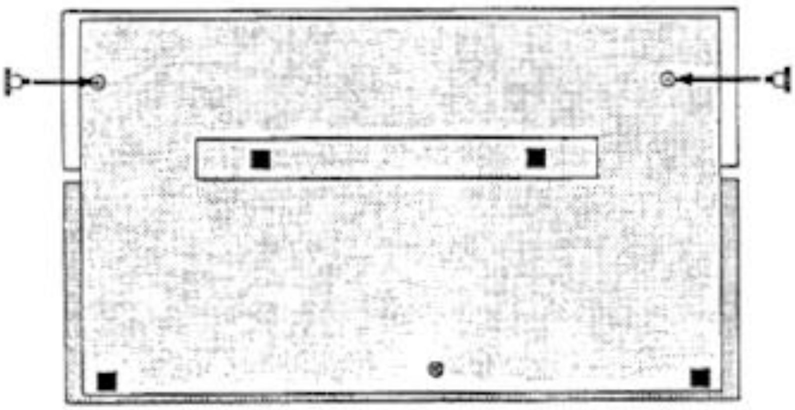
Remove the connector cover on the back of the CE-150 and store it on the bottom of the CE-150.



Insert the CE150 / MC-150 into the guide pin, turning it counterclockwise relative to the mounting plate (Fig. 1). Turn it clockwise until the hooks engage properly. Visually inspect.



Carefully insert the MC-12(A) connector into the CE 150 socket so that the threaded holes are centered (Fig. 3). Screw the knurled screws in vertically (do not tilt; the tension is required!).



**Commissioning the MC-12(A)**

- Turn off the PC-1500(A)
- Remove the connector cover on the back of the CE-150 and store it on the bottom of the CE-150.
- Plug the PC-1500(A) onto the CE-150
- Turn on the PC-1500(A). The MC-12(A) is initialized when the device is turned on for the first time. If the MC-12(A) is not disconnected from the CE-150, the settings and measured values from the last measurement will be retained even after the PC-1500(A) is turned off.
- Please also observe the general operating instructions for the correct connection of the power adapter.

# Chapter 4

## MULTIMETER

The CMOS operating system includes a program for operating the MC-12(A) as a multimeter. The various functions of this multimeter are activated by pressing individual buttons, just like a conventional digital multimeter.

In addition to the standard digital display of measured values, the MC-12(A) multimeter features a graphic analog display that shows the minimum and maximum signal levels for varying signals. The voltage at each of the five analog inputs can be measured either within the set range (with analog display) or with automatic ranging by the MC-12(A) (pure digital display). If the measuring range is exceeded, "OVERFLOW" is displayed.

Each input channel can be assigned a numeric function for converting the measured values using the SETFUNCTION command (see Section 7.3). The values of this function can be displayed on the multimeter.

If the graphic analog display is activated, all measurements are performed with an 8-bit resolution. 256 individual measurements (approximately 80μs per individual measurement) are taken within an interval of 20ms, followed by the average, minimum, and maximum values. With a purely digital display of the measured values, if an 11-bit module is present, the measurements are performed with an 11-bit resolution. The multimeter program is started by entering the MULTIMETER <ENTER> command. However, the MC-12(A) must first be switched on with the MCON <ENTER> command.

**Measurement example**

- Switch on the analog part of the MC-12(A) using the command: MCON <ENTER>

- Set the input function "4*CHA(3)" on channel 3:
- SETFUNCTION 3, 4*CHA(3) <ENTER>

- Start the multimeter with the command: MULTI.<ENTER>

- The display shows ‘CHANNEL 1= 0.000V’
- Apply a voltage of +2V to channel 3.
- To switch to channel 3, press the <3> button.
- The display shows 'CHANNEL 3= 2.01V'.
- Press the P key and the display will be printed along with the values of all other channels on the CE-150.
- To display the input function, press the <DEF> key.
- The display shows 'CHANNEL 3= 8.04’
- Apply a 2V AC voltage of 50 Hz to channel 1.
- Press the <1> button.
- Press the <N> key.
- A bar appears on the display that ranges from the minimum to the maximum of the input voltage.
- Press the <E> key. Multimeter operation will end.

**Function of the multimeter**

The multimeter program is called using the MULTIMETER command. The analog component of the MC-12(A) must be turned on; otherwise, the error message "ERROR 101" appears (see Chapter 6). If the system's battery voltage is too low, the error message "ERROR 135" appears.

In multimeter mode, the PC-1500(A) display continuously shows the current measured value for the respective input channel. By pressing the indicated keys, you can initiate the functions described below. Please note that the keys sometimes need to be held for a brief moment before the desired function is initiated.

Key: <1>…<5>

Switches the display to inputs 1…5

Key: <A>

Enabling automatic range selection (AUTO RANGE): In this mode, the MC-12(A) automatically selects the most sensitive measuring range for each measurement. The measured value is displayed digitally only. If the corresponding module is present, measurements are performed with 11-bit resolution.

Key: <N>

Switch on normal display: Automatic range selection is deactivated, and the measurement is taken in the currently selected range. At the same time, the analog display is switched on. If a pure DC voltage signal is not present, a bar graph appears, ranging from the minimum to the maximum value of the signal during one measurement period (20ms). Measurements are generally performed with 8-bit resolution.

Key: <H>

Hold Minimum and Maximum: Like <N>, but additionally, the analog display shows the absolute minimum and maximum of the input signal since the last time the <H> key was pressed.

Key: <H>

Reduce measuring range: The measuring range of the currently displayed input is halved. This function is only effective when automatic measuring range selection is disabled (in operating modes N and H).

Key: <H>

Increase measuring range: Doubles the measuring range of the currently displayed input. This function is only effective if automatic measuring range selection is disabled.

Key: <DEF>

Switching the function value display on/off: Each input channel can be assigned a function for converting the measured values using the SETFUNCTION command (see Section 7.3).

Key: <P>

Print: The measured values of all five input channels are printed on the CE-150 plotter

Key: <E>

End of multimeter operation, return to the calling BASIC program.

Key: <BREAK>

Aborting the multimeter operation and the calling BASIC program.

# Chapter 5

## TRANSIENT RECORDER

The MC-12(A) operating software includes a program for transient recording.

This program can only be used if no other BASIC program is stored in the PC-1500(A)'s memory. Therefore, save any programs stored in memory to a cassette tape and clear the program memory in PRO mode (press the MODE key) by entering the NEW <ENTER> command. Switch the calculator back to RUN mode (press the MODE key again) and enter TRANSREC <ENTER>.

This command loads the MC-12(A) transient recorder program into the PC-1500(A). If ERROR 102 is displayed, the PC-1500(A)'s program memory is still occupied by another program.

### Measurement example

- Load the program with the command: TRANS. <ENTER>
- Apply a sine signal with a frequency of 00Hz and a peak voltage of +-20mV to channel 1.
- For example, apply a DC voltage of 1V to channel 2.
- Start the transient recorder with RUN <ENTER>
- Display: HOW MANY CHANNELS (1..5)?
  Input: 2
  Two channels are scanned.
- Advertisement: HOW MANY BLOCKS
  BUFFERLENGTH (1..13)?

Input: 3 <ENTER>
A buffer of three blocks of 256 values is created for each of the two input channels. Therefore, 768 pairs of values are recorded during the measurement.

Display: SELECT
TRIGGER CHANNEL (1..5)?
Input: 1
Channel 1 triggers the measurement process and is the first channel sampled.

Display: SELECT
NEXT CHANNEL (1..5)?
Input: 2
The second detected channel is channel 2.

Display: SELECT RANGES!
DIRECT VIEW WITH
MULTIMETER (Y/N)?
Input: N
If you enter "Y," the measuring ranges can be set in multimeter mode using the level display in M or H mode (see Chapter 4). Now, however, the ranges are set directly:

Display: RANGE OF
CHANNEL 1 (0...4.88)
Input: 0.02 <ENTER>
Channel 1 is set to the 38mV range. (This is the most sensitive range, containing 0.02V.)

Display: RANGE OF
CHANNEL2 (0...4.88):?
Input: 1 <ENTER>
The 1.23V range is set on channel 2.

Display: RANGES OK (Y/N)?
Input: Y

Display: HOW MANY BLOCKS
PREHISTORY (0..2):?
Input: 0 <ENTER>
No history (signal curve before the trigger time) is recorded.

Display: SELECT SCANTIME (ms)
SCANTIME (0.25 ...):?
Input: 0.25 <ENTER>
The specified minimum scan time (0.25ms) is selected.

Display: TRIGGER
LEVEL (0...0.038):?
Input: .01 <ENTER>
The trigger level (channel 1) is set to 0.01V

Display: TRIGGER EDGE (+/-):
Input: <+>
Positive edge triggering is selected, i.e., the measurement starts when the 0.01V level on channel 1 is crossed for the first time in the positive direction.

Display: PRESS ENTER TO START SCAN
Input: <ENTER>
Press ENTER when you are ready to take a measurement.

Display: SCANNING... (briefly during the measurement)
The MC-12(A) now waits until the trigger condition is met.
Afterward, a pair of values from inputs 1 and 2 is transferred to buffers 1 and 2 every 0.25ms until 768 pairs of values have been acquired.

Display: SCREEN DISPLAY (Y/N)?
Input: N
By entering "Y," the signal waveform can be viewed on the oscilloscope screen when an oscilloscope is connected. The signal waveform display can be stretched or compressed on the screen, and the screen display can also be copied to the plotter (see Section 11.1).

Display: PLOT (Y/N)?
Input: Y
The entire acquired signal waveform is displayed on the plotter.

Display: NEW SCAN (Y/N)?
Input: N
If "Y" is selected, a new measurement is taken with the same set parameters.

Display: NEW PARAMETERS (Y/N)?
Input: N
This terminates the program. Entering "Y" starts the dialog again.

As the example shows, the transient recorder program guides you through all the steps necessary for a measurement in a dialog.

If the program requests an input, the possible range of values is always given. If an input error occurs, a short beep sounds and the corresponding question is repeated by the program.

After starting the program, the following question appears on the display:

HOW MANY CHANNELS (1..5):

whereupon you enter the desired number (1...5) of input channels to be sampled. A buffer is created for each input channel. Therefore, if two channels are to be sampled, the program creates two buffers.

The following question will then appear on the display:

HOW MANY BLOCKS
BUFFERLENGTH (1..nn)

Now you can specify how many values per channel should be captured during the sampling process. A minimum of 256 values will be captured, and a maximum of nn such blocks, with nn varying between 28 and 5 depending on the number of channels to be sampled (see Chapter 9.1).

The display next shows:

SELECT
TRIGGER CHANNEL (1..5):

Then enter the number of the input channel that will serve as the trigger input. The measured values from this input are stored in the first buffer.

If more than one channel is to be scanned, the following prompt will appear:

SELECT
NEXT CHANNEL (1, 5):

whereupon you enter the number of the next input channel to be sampled, which is then stored in the 2nd buffer.

This is repeated until all input channels to be scanned are selected. The following appears on the display:

NOW SELECT RANGES!

DIRECT VIEW WITH
MULTIMETER (Y/N)?

If this question is answered with "Y," the MULTIMETER program (see Chapter 4) is called up. This allows the control of the individual channels to be set using the bar graph on the display.

Otherwise, the measuring ranges of the selected channels are displayed after the question:

RANGE OF
CHANNEL n (0...4.88):?

by entering the upper range limit (signal level in VSS).

This question is repeated until a measuring range is assigned to each selected channel.

After all measuring ranges have been determined, the question:

HOW MANY BLOCKS
PREHISTORY (0..nn):?

specifies how many values from prehistory, i.e., the signal curve before triggering, should be recorded. The number of values is specified in blocks of 256 values.

Since the history must always be one block (of 256 measured values) shorter than the entire buffer memory, no history can be recorded with a buffer length of one block. The maximum possible length nn of the history is displayed.

SELECT SCANTIME (ms)
SCANTIME (x.xxx ...):?

The sampling interval can now be set, with the minimum sampling time x.xXxx being displayed. Depending on the number of selected channels, their sampling ranges, and the length of the history, this minimum time varies between 0.033 ms and 0.550 ms. The maximum possible sampling time is over 3200 ms.

For the questions:

TRIGGER
LEVEL (0...x.xxx):

and

TRIGGER EDGE (+/-):

The desired values for the trigger level and edge trigger type must be specified. The trigger level must be within the measurement range of the trigger channel.

By confirming the message

PRESS ENTER TO START SCAN

Press the <ENTER> key to stop the measurement. For verification, the following appears on the display:

SCANNING...

Now the system waits (while simultaneously recording the history) until the trigger condition is met. Subsequently, the measured values from the input channels are read into the buffer memory at the specified time interval.

If the trigger condition is not met (e.g. trigger level is too high), the waiting state of the MC-12(A) can be aborted using the <BREAK> key.

You can then answer the following questions:

SCREEN DISPLAY (Y/N)?
PLOT (Y/N)?
NEW SCAN (Y/N)

The SCREEN menu can be called up to display the acquired signals on the oscilloscope (see section 11.1), a display of the signal curve can be triggered on the plotter, or the measurement can be repeated. If the question

NEW PARAMETERS (Y/N)?

If the answer is "N," the program terminates, retaining all stored values. Otherwise, the program execution is repeated.

# Chapter 6

**CMOS OPERATING SYSTEM**

The following chapters describe the commands implemented in PC-1500(A) BASIC for operating the MC-12(A).

These commands can be used to control all MC-12(A) operations:

- Turning the MC-12(A) on/off and initializing
- Individual measurements
- Control via switches, relays, and D/A converters
- Recording of signal waveforms in buffer memory (transient measurements)
- Processing the stored signals
- Outputting the stored signals on the plotter or oscilloscope

At the beginning of each of the following chapters you will find a short introduction explaining how the commands work and special features to be observed.

Examples of using the commands are provided at the end of each chapter.

The individual commands are explained in turn within each chapter.

You can use the CMOS BASIC commands in your own BASIC programs, or you can enter and execute them directly.

However, this assumes some knowledge of BASIC and the operation of the PC-1500(A). You can find everything you need in the PC-1500(A) manual.

The individual CMOS commands are described in this manual according to the following scheme:

**Syntax**:

The command and the required parameters are listed, including the permissible range of values for the parameters.

Example: SETRANGE <channel number>,<range limit>

Abbr.: SETR.
Channel number: 1 ... 5
Range limit: 0 ... 4.88

This means that when using SETRANGE, a valid channel number and a range limit must always be specified.

| | | |
|---|---|---|
| So: | SETRANGE 1,2.5 | is allowed |
| | SETRANGE 1<br>(missing range limit) | is not allowed |
| | SETRANGE 7,4<br>(channel number too large) | is not allowed |

For all numeric parameters, any BASIC expression can be used instead of simple numbers.

Example:

```
N=1
R=0.5
SETRANGE N,2*R
```

If only whole numbers are important (e.g. channel numbers), the fractional part is simply cut off - as is usual with the PC-1500(A).

Example: SETRANGE 2.1,4 is equivalent to SETRANGE 2,4

Like all PC-1500(A) BASIC commands, CMOS commands can be abbreviated using a period; the shortest permissible form is indicated under 'Abbr.:'.

Example: SETR.2,4 is equivalent to SETRANGE 2,4

**Initialization values:**

For commands that specify certain system parameters, the initialization values of these parameters are given below, as they are set when the system is first switched on or after the INIT command.

All parameters you have set once remain effective (even after switching the system on/off) until they are changed with corresponding commands.

**Error messages:**

If incorrect commands are issued, the PC-1500(A) will report this with:

ERROR nnn IN mmm (during program execution)
or ERROR nnn (in direct mode)

where nnn is the respective error number and mmm is the program line

The specific error messages for the CMOS commands are listed after the command description.

The CMOS error numbers are all in the range 100...139.

However, the general error messages of the PC-1500(A)-BASIC, such as "ERROR 1" (syntax error), are not listed. If such errors occur, you must consult the PC-1500(A) manual.

## 6.1 (RE-INITIALIZATION / SWITCHING ON - OFF THE MC-12(A))

The commands described in this section can be used to initialize the MC-12(A) system, turn it on/off to reduce power consumption, or put it into sleep mode.

Caution: Before measurements, the analog component of the MC-12(A) must be switched on using the MCON command; otherwise, the error message "ERROR 101" will be displayed. Pressing the <ON> key only switches on the PC-1500, not the MC-12(A).

**INIT**

Abbreviation: **INI.**

Initials the measuring system and sets the initial values for all system parameters as specified in the following sections. At the same time, the MC-12(A) is turned on.

**INIT B**
**INIT U**

Abbreviation: **INL.B INIL.U**

INIT B sets the MC-12(A) to bipolar mode, **INIT U** to unipolar mode, otherwise as described for INIT.

Please note, however, that changing the operating mode requires recalibration of the MC-12(A) (see Appendix C).

When the MC-12(A) is first switched on after being connected, bipolar mode is automatically initialized.

**MCON**
**MCOFF**

Abbr.: **MC. MCOF.**

To reduce power consumption, the analog section of the MC-12(A) can be turned on and off using these BASIC commands.

If a command is issued while the analog section is turned off that can only be executed when the unit is turned on, the error message "**ERROR 101**" is displayed.

If the system's battery voltage is too low when the MC-12(A) is turned on, the error message "**ERROR 135**" is displayed.

The MCON command is delayed by approximately 0.5 seconds, while **MCOFF** takes effect immediately.

Error messages: **ERROR 135**: Battery voltage too low.

**SLEEP** <rest time>

Abbreviation: **SL.**

Rest time: 2 … 65535 (seconds)

This command puts the entire system into a power-saving sleep state for the specified time. The sleep time is specified in seconds.

The PC-1500(A) remains in sleep mode for the specified time, while the analog component of the MC-12(A) is turned off. Power consumption in sleep mode is only about 5mA, compared to approximately 20mA when running a BASIC program with the MC-12(A) turned off.

Sleep mode can be interrupted using the <BREAK> button.

Example: **SLEEP 10**

Error messages: **ERROR 19**: Invalid sleep time.

# CHAPTER 7

## INDIVIDUAL MEASUREMENTS

For individual measurements, the measured values from the analog inputs are transferred directly to a BASIC program or (in manual mode) displayed. Depending on the BASIC program, a maximum of 10 measurements per second can be performed.

The measuring range can be selected independently for each of the five analog inputs. 11 measuring ranges are available for each of the five analog inputs (see table on page xx).

In **AUTORANGE** mode, the MC-12(A) automatically selects the most sensitive measuring range for each individual measurement; the range specifications are temporarily overridden.

For each measurement, the system automatically performs 256 individual measurements within 20 milliseconds, and the average value is calculated. If the measurement range is exceeded, the error message "**ERROR 110**" is displayed.

If the MC-12(A) is equipped with the MC-12.1 (11-bit extension module), individual measurements in the corresponding measuring ranges (see table in Chapter xx) are carried out with 11-bit accuracy, otherwise with 8-bit.

All measured values or measuring ranges are specified in "volts" within the BASIC commands, the input channels are addressed via the channel numbers 1...5.

## 7.1 COMMANDS FOR PARAMETER SETTING

**SETRANGE** <channel number>, <range limit>

Abbr.: **SETR.**

Channel number: 1...5
Range limit: 0 ... 4.8 (volts)

This command defines the measuring range of the individual input channels.

From the 11 possible measuring ranges (see table in Chapter xx), the one with the finest resolution that contains the specified range limit is selected.

Initialization: 4.88V range on all channels

Example: **SETRANGE 1,2**
sets the 2.45V range on channel 1.

Error messages: **ERROR19**: Invalid channel number
**ERROR 110**: Invalid range limit

**RANGE** <channel number>

Abbreviation: **RANK.**

Channel number: 1...5

This control function returns the range limit set on the specified input channel.

Example: **RANGE 1**

Error message: **ERROR 19** : Invalid channel number.

**AUTORANGE ON**
**AUTORANGE OFF**

Abbr.: **AU.O. AU.OF.**

These commands switch the automatic measuring range selection on/off.

If this is switched on, the measuring range specified by **RANGE** is ignored for individual measurements and the measuring range is automatically determined for each measurement by refining the range step by step.

For low input voltages, the measurement process takes significantly longer when using automatic ranging.

After automatic ranging is disabled, the previously set range limit becomes active again.

Initialization: **AUTORANGE ON**

## 7.2 MEASUREMENT COMMANDS

**CHA** <channel number>

Abbr.: **CH.**

Channel number: 1...5

This function returns the current measured value at the input channel specified by the channel number.

With **AUTORANGE OFF** the measurement is carried out in the set measuring range; with **AUTORANGE ON** the most sensitive measuring range is automatically selected.

The CHA() function is used like any other BASIC function within numeric expressions.

Example: **A=CHA (1)**

Error messages:

**ERROR 19**: Invalid channel number
**ERROR 110**: Input voltage outside the measuring range
**ERROR 101**: MC-12(A) is turned off
**ERROR 135**: Battery voltage too low

**INCHA** <channel number>,<variable name>

Abbr.: **INC.**

Channel number: 1...5
Variable name: numeric BASIC variable such as **A, B1, XX(2)**

This command assigns the corresponding input value to the specified numeric variable. Except for the different syntax, this command behaves exactly like **CHA().**

Example: **INCHA 1,B**

Error messages: as for **CHA ()**

## 7.3 FUNCTION

Each of the input channels can be assigned a fixed numerical function for converting the measured values.

This simplifies the scaling of measured values, characteristic curve linearization, and unit conversion.

The values of the respective function can be displayed in the **MULTIMETER** (see Chapter 4).

**SETFUNCTION** <channel number>,<function>

Abbreviation: **SETF.**

Channel number: 1...5

Function: Numeric expression (must not occupy more than 40 bytes of memory)

The specified channel is assigned the respective function, generally a numeric expression with **CHA(I)** as an argument.

The function expression is calculated internally immediately in order to generate any error messages during the function definition.

The respective function is evaluated in subsequent **INFUNCTION** commands or in the **MULTIMETER** program.

Initialization: **CHA** (<channel number»>) on all channels

Example: **SETFUNCTION 1,3*CHA(1)+5**

Error messages: **ERROR 119**: Function expression too long.
Otherwise, e.g., error messages from **CHA()**

**INFUNCTION** <channel number>,<variable name>

Abbreviation: **INF.**

Channel number: 1...5

Variable name: numeric BASIC variable such as **A, XY**, ...

The specified variable is assigned the current value of the function defined with the SETFUNGCTION command.

Example: **INFUNCTION 1,C**

Error messages: Depend on the function

## 7.4 EXAMPLES OF INDIVIDUAL MEASUREMENTS

Apply a voltage of 2V to input 1 and enter the following commands:

| Input | Display | Notes |
|---|---|---|
| **`MCON`** | > | Turning on the MC-12 |
| **`SETRANGE 1,2`** | > | Measuring range 2V |
| **`RANGE(1)`** | 2.45 | Measuring range limit |
| **`AUTORANGE OFF`** | > | Auto range selection switched off |
| **`CHA(1)`** | 2 | Current measured value on channel 1 |
| **`A=3*CHA(1)`** | 6 | |
| **`A`** | 6 | |
| **`INCHA 1,B`** | > | The current input value is assigned to variable B. |
| **`B`** | 6 | |
| **`SETF. 1,2*CHA(1)+15`** | > | Define a function for subsequent **`INFUNCTION`** command. |
| **`INFUNCTION 1,C`** | 19 | Transform the current measured value using the defined function. |
| **`C`** | 19 | |

**Example program: Simple data logger**

The following program records the voltage at input 1 on the printer every full minute.

To enter the program, switch the PC-1500(A) to PRO mode. After entering the program, you can start the program in RUN mode using the RUN command. The program is aborted with <BREAK>.

After the first minute, every minute is logged:

> 1st MIN: 2.01V
> 2nd MIN: 2.01V

For information on BASIC commands and the TIME function, refer to the PC-1500(A) instruction manual.

| Program: | Note: |
|---|---|
| `10 I=1` | |
| `20 MCON:AUTORANGE ON` | |
| `30 T=TIME` | Time |
| `40 IF INT(T*100 )/100<>T GOTO 30` | Wait for full minutes |
| `50 LPRINT I;" . MIN: ";CHA(1);"V"` | Print |
| `60 SLEEP 57` | Delay (Sleep) |
| `70 I=I+1:GOTO 30` | |

**Example program: Minimum, Maximum, Average**

The following program calculates the average, minimum, and maximum values for over 100 measurements on channel 1.

To enter the program, switch the PC-1500(A) to **PRO** mode. After entering the program, you can start the program in **RUN** mode using the **RUN** command.

After completing the 100 measurements, the minimum, average, and maximum values are printed:

| | | |
|---|---|---|
| MIN: | MID: | MAX: |
| 2.01 | 2.45 | 2.61 |

| Program: | Note: |
|---|---|
| `10 MI=10:MA=-10:MW=0` | initialization |
| `20 MCON` | |
| `30 FOR I=1 TO 100` | |
| `40 A=CHA(1)` | |
| `50 IF A<MI LET MI=A` | minimum |
| `60 IF A>MA LET MA=A` | Maximum |
| `70 MW=MW+A` | Mean |
| `80 NEXT I` | |
| `90 MCOFF` | |
| `100 LPRINT " MIN: MIT: MAX:"` | |
| `110 USING "##.##"` | |
| `120 LPRINT MI;MW/100;MA` | formatting |
| `130 END` | |

# CHAPTER 8

## CONTROL

The four analog switches and two D/A converters of the MC-12 system, as well as the two remote relays of the CE-150, can be used for control. However, the following restrictions apply:

If the analog section of the MC-12(A) is turned off, the D/A converters and analog switches are disabled. The last preset state is only restored when the MC-12(A) is turned on again.

Attention: The MC-12(A) is also switched off during the **PLOT** or **HARDCOPY** command (see Chapter 11).

D/A converters are also used to display signals on the oscilloscope; they cannot be used simultaneously for control purposes.

**SWITCH** <switch number> **ON**
**SWITCH** <switch number> **OFF**

Abbreviation: **SW.**

Switch number: 1...4

Turns the specified analog switches on/off. Ineffective during **MCOFF** or **PLOT**.

Example: **SWITCH 1 ON**

Error messages:
**ERROR 19**: Invalid switch number
**ERROR 101**: MC-12(A) switched off
**ERROR 135**: Battery voltage too low

**RELAY** <relay number> **ON**
**RELAY** <relay number> **OFF**

Abbreviation: **REL**.

Relay number: 0 or 1

Turns the remote relays '**REM0**' and '**REM1**' on the CE-150 on/off. Relay 0 is only activated when the 'REMOTE' switch on the CE-150 is in the 'ON' position.

Initialization: Both relays off

Example: **RELAY 0 ON**

Error messages: **ERROR19**: Invalid relay number

**OUTCHA** <Output channel>,<Voltage>

Abbr.: **OU.**

Output number: 1 or 2
Voltage: -5...+4.96 (volts, bipolar operation)
±0...+4.98 (volts, unipolar operation)

Sets the specified voltage to the respective analog output. Ineffective during **MCOFF**, **PLOT**, or oscilloscope operation.

Initialization: 0V at both outputs

Example: **OUTCHA 1, 2, 5**

Error messages:

| | |
|---|---|
| **ERROR19**: | Invalid output number |
| **ERROR110**: | Invalid voltage |
| **ERROR101**: | MOC-12(A) switched off |
| **ERROR135**: | Battery voltage too low |

# CHAPTER 9

## MEASURING TRANSIENTS

To capture fast events (transients) with a maximum of 30,000 measurements per second, the measured values are automatically read into the buffer memory at a specified time interval and only then evaluated.

For this purpose, you can create a fixed number of buffer memories in the MC-12(A), while the program memory of the PC-1500 remains unaffected.

The buffer size can be freely selected in blocks of 256 values each, ranging from 1*256 to 28*256 (=7168) values.

Reading the measured values from the analog inputs into the buffers is controlled by special commands that specify a trigger level for starting the reading, the sampling interval, the number of channels to be sampled, and the length of the history to be recorded.

The measuring range set at the inputs using **`SETRANGE`** is also taken into account for transient measurements; however, automatic range selection is not possible.

Transient measurements are generally performed with 8-bit resolution. The maximum achievable sampling rate for transient measurements depends on the number of channels to be sampled, the different measurement ranges, and the length of the history to be recorded.

## 9.1 Buffer memory management command

Up to 26 buffer memories can be created in the MC-12(A) to store signal waveforms.

When measuring transients, the measured values are stored as 8-bit numbers in the buffers. The set measurement range and the respective sampling parameters are also recorded in the buffers.

Each buffer can store exactly one input signal. The acquired signal duration is the product of the selected sampling interval and the buffer length.

To ensure that 11-bit values can be stored in the buffers with full precision when an 11-bit expansion module is present, it is possible to create double-precision buffers. However, signals with 11-bit resolution can only be stored using single measurements and **BUFWRITE** (see Section 10.3), not with automatic transient measurements.

**BUFINIT** <buffer count>
**BUFINIT** <buffer count>,<buffer size>

Abbreviation: **BUFTIL.**

Buffer count: 1…6
Buffer size: 1…28 (blocks of 256 8-bit values)

This command creates the desired number of buffers. At the same time, the buffers are deleted.

If no buffer size is specified, the buffers will be given the maximum possible size as listed in the table below.

Otherwise, the buffer size can be specified in blocks of 256 values.

**Possible buffer sizes for 8-bit buffers**

| Number | Length (blocks) | Length (values) |
|---|---|---|
| 1 | 1 ... 28 | 256 ... 7168 |
| 2 | 1 ... 13 | 256 ... 3328 |
| 3 | 1 ... 8 | 256 ... 2048 |
| 4 | 1 ... 6 | 256 ... 1536 |
| 5 | 1 ... 5 | 256 ... 1280 |
| 6 | 1 ... 4 | 256 ... 1024 |
| 7 ... 9 | 1 ... 3 | 256 ... 768 |
| 10 ... 13 | 1 or 2 | 256 or 512 |
| 14 ... 26 | 1 | 256 |

Initialization: 5 buffers of 5 blocks (1280 values)

Example: `BUFINIT 3.2` (3 buffers, 512 values)

Error messages: **`ERROR 19`** Invalid number of buffers
**`ERROR 121`** Invalid buffer size

**`DBUFINIT`** number_of_buffers
**`DBUFINIT`** number_of_buffers, buffer_size

Abbreviation: **`BUFI.`**

number of buffers: 1 ... 13
Buffer size: 1 ... 14 (blocks of 256 16-bit values)

This command makes it possible to create double-precision buffer memories that can hold 16-bit values. This is not used for transient measurements, as these are always carried out with 8-bit resolution. For writing to the buffer memory by BASIC programs (see section 10.3), however, this possibility is very useful in connection with 11-bit direct measurements.

This command is completely analogous to **`BUFINIT`**, the maximum buffer size also depends on the number of buffers selected, as can be seen in the following table:

**Possible buffer sizes for 16-bit buffers**

| Number | Length (blocks) | Length (values) |
|---|---|---|
| 1 | 1 ...14 | 256 ... 3584 |
| 2 | 1 ... 6 | 256 ... 1536 |
| 3 | 1 ... 4 | 256 ... 1024 |
| 4 | 1 ... 3 | 256 ... 768 |
| 5 | 1 ... 2 | 256 ... 512 |
| 6 | 1 ... 2 | 256 ... 512 |
| 7 ... 13 | 1 | 256 |

Example: **`DBUFINIT 2.2`** (2 16-bit buffers per 512 values)

Initialization: basically 8-bit buffer, see **`BUFINIT`**

Error messages: **`ERROR 19`** Invalid number of buffers
**`ERROR 121`** Invalid buffer size

**`BUFNUM`**
**`BUFLEN`**

Abbreviation: **`BUFN. BUFL.`**

These control variables contain the number and length (number of values) of the buffer memory.

Examples: **`D=BUFNUM`**
**`E=BUFLEN`** (number of values)
**`F=BUFLEN/256`** (number of blocks)

## 9.2 COMMANDS FOR SETTING THE SCANNING PARAMETERS

**`SETPREHIST`** prehistory_blocks

Abbreviation: **`SETP.`**
prehistory_blocks: 0 ... buffer size-1 (blocks)

This command is used to specify how many blocks of 256 values are to be used in each buffer to store the previous history, i.e. the signal curve before the trigger condition occurs.

However, the maximum sampling rates can only be achieved with a pre-shift length of 0 or 1 blocks.

Initialization: 0 Blocks

Example: **SETPREHIST 1** (256 values history)

Error messages: **ERROR 19** Illegal number of blocks

**PREHIST**

Abbreviation: **PRE.**

This control variable contains the number of selected history blocks.

Example: **D=256*PREHIST** (number of values for previous events)
**E=2.56*PREHIST/BUFLEN** (history in % of the total signal detected)

**SETTRIGGER** level

Abbreviation: **SETT.**

level: -4.92...4.88 (Volt, bipolar operation)
0...4.92 (Volt, unipolar operation)

This specifies a trigger level that triggers the start of automatic reading of the measured values after the INSCAN command (see below).

Whether the trigger level is within the measuring range of the trigger channel is only checked with the INSCAN command.

Initialization: Trigger level 0V

Example: **SETTRIGGER 0.5**

**TRIGGER**

Abbreviation: **TRI.**

This control variable contains the currently set trigger level.

Example: **G=TRIGGER**

**SETSCANTIME** sampling_interval

Abbreviation: **SETS.**

sampling interval: 0 ... 3.27 (seconds)

This command is used to specify the length of the sampling interval for the automatic reading of the measured values in.

The maximum achievable sampling rate depends on the number of inputs to be sampled, their measuring ranges and the length of the previous history.

When **SETSCANTIME 0** is entered, the shortest scanning interval possible with the currently set parameters is selected.

If the value of the scanning interval is too low, the error message "**ERROR 111**" is only issued when the **INSCAN** command is executed.

The sampling time can be specified in 50μs increments, the minimum value can be found in the following list:

| **Condition** | **Minimum scanning time** |
|---|---|
| 1 channel<br>History: 0...1 blocks | 33us |
| 1 channel<br>History: Any | 50us |
| k Channels (V=2...5)<br>History: 0...1 blocks<br>all channels same measuring range | K * 50us |
| k Channels (V=2...5)<br>History: Any<br>Different measuring ranges | 50 + k * 100us |

Initialization: Sampling interval 0.001s

Example: **SETSCANTIME 50E-6**

**SCANTIME**

Abbreviation: **SCANT.**

This control variable contains the set sampling time.

Example: **L= SCANTIME*BUFLEN** (total signal duration)

**SELECT** +channel number, buffer number
**SELECT** -channel number, buffer number
**SELECT** +channel number 1, buffer number 1;channel number 2, buffer number 2
**SELECT** -channel number 1, buffer number 1;channel number 2, buffer number 2

Abbreviation: **SE.**

channel number: 1 ... 5
buffer number: 1 ... Number of buffers

Selects the specified input channels for subsequent transient measurements and assigns buffer memories to them.

The first channel specified in the **SELECT** command is selected as the trigger channel, whereby negative edge triggering is set with the negative sign and positive edge triggering with the positive sign.

Initialization: **SELECT +1,1**

Example: **SELECT -1,1;2,2**

| | | |
|---|---|---|
| Error messages: | **ERROR 19** | Invalid channel or buffer no. |
| | **ERROR 130** | Too many channels selected |
| | **ERROR 131** | Buffer memory selected twice |

## 9.3 MEASUREMENT COMMAND

**INSCAN**

Abbreviation: **INS.**

This command triggers a transient measurement:

After checking the set parameters, the system waits until the specified trigger condition is met. The system waits until the trigger level is crossed for the first time in the specified direction (+/-).

The measured values are then read from the selected inputs into the associated buffer memory at the specified intervals until the buffer memory is full.

While the system is waiting for triggering, the previous history is recorded. If the trigger occurs too early, the prehistory is shortened accordingly, i.e. fewer prehistory values than specified are recorded.

Once the measurement is complete, a zero-point drift correction is carried out using the stored values.

Pressing the <BREAK> button once aborts the measuring process, whereby the BASIC program is only aborted when the <BREAK> button is pressed a second time.

| | | |
|---|---|---|
| Error messages: | **ERROR 110** | Trigger level outside the measuring range of the trigger channel |
| | **ERROR 111** | Specified sampling time too low |
| | **ERROR 101** | MC-12 switched off |
| | **ERROR 135** | Battery voltage too low |

**`INSCANN`**

Abbreviation: `INS.N`

After entering this command, the measurement begins immediately. Unlike the normal **`INSCAN`** command, there is no waiting for a trigger condition. The number of channels and their buffer allocation correspond to the settings specified with the **`SELECT`** command.

Error messages: as with **`INSCAN`**

## 9.4 EXAMPLES FOR TRANSIENT MEASUREMENTS

If a signal is to be recorded on channel 1, the following commands must be issued:

| Input | Display | Notes |
|---|---|---|
| **`MCON`** | > | Switching on the measuring system |
| **`SETRANGE 1,2.4`** | > | Range limit for channel 1 |
| **`BUFINIT 5,4`** | > | 5 buffer storage with 4 blocks |
| **`BUFLEN 1024`** | | Buffer length is 1024 measured values |
| **`SELECT +1,1`** | > | Channel 1 as trigger channel<br>Store signal in buffer 1 |
| **`SETPREHIST 1`** | > | Prehistory 1 Block |
| **`SETTRIGGER 0.3`** | > | Trigger level 0.3V |
| **`SETSCANTIME 50E-6`** | > | Sampling interval 50us |
| | | |
| **`INSCAN`** | > | Start the measurement<br>Standby symbol " >" appears after the trigger condition is met<br>and end the measurement |
| **`SELECT +1,2`** | > | Next signal in buffer 2 |
| **`INSCAN`** | | Next measurement |

**Example program: Recording 10 signals on channel 1**

Switch the PC-1500 to PRO mode and enter the program below.

After starting this program in RUN mode, 10 transients are recorded on channel 1 with the sampling parameters set in the program.

| Program: | Notes: |
|---|---|
| `10 WAIT 32:BUFINIT 10,2` | 10 buffer memory with 512 values |
| `20 SETRANGE 1,0.5` | Signal level 0.5 VSS |
| `30 SETSCANTIME 50E-6` | Sampling interval 50µs |
| `40 SETTRIGGER 0.1` | Trigger level 0.1V |
| `50 SETPREHIST 1` | History 256 values |
| `60 MCON` | |
| `80 FOR I=1 TO 10` | |
| `90 PRINT "MEASURMENT";I` | |
| `100 SELECT +1,I` | Channel 1 -> Buffer I |
| `110 INSCAN` | Measurement |
| `120 BEEP 1` | Beep |
| `130 NEXT 1` | |
| `140 MCOFF` | |

# CHAPTER 10

## ACCESS TO THE BUFFER MEMORY

The measured values stored in the buffers can be easily read by a BASIC program.

In the same way, values can be written to the buffer memory by a BASIC program for later output on the oscilloscope or plotter.

Please note that the numerical values in the buffers are saved in the same format as the measured values.

The value range (analogous to the measuring range) must therefore be specified before values are written to the buffers. If this range is exceeded, an error message is issued.

The values are normally saved with a resolution of 8 bits, unless double-precision buffers are used.

Therefore, the values that are written to the buffers by a BASIC program should exhaust this value range as far as possible (just as measurements should be controlled as well as possible).

If this is not observed, it can happen, for example, that all values written to the buffers within the 8-bit resolution result in NULL.

## 10.1 READING FROM THE BUFFER MEMORIES

**BUFREAD** buffer_number, position, variable_name

| | |
|---|---|
| buffer_number: | 1 ... number of buffers |
| position: | 1 ... Buffer length |
| variable_name: | Any BASIC variable such as **A, XY, B(2)** |

The measured value stored in the respective buffer at the specified position is assigned to the specified variable.

Example: **BUFREAD 1,100,A**

| | | |
|---|---|---|
| Error messages: | **ERROR 19** | Invalid buffer number |
| | **ERROR 121** | Non-existent position |

## 10.2 WRITING TO THE BUFFER MEMORY

**BUFOPEN** buffer_number
**BUFOPEN** buffer_number, range_limit

Abbreviation: **BUFO.**

| | |
|---|---|
| buffer_number: | 1 ... number of buffers |
| range_limit: | 0 ... 4.88 |

Deletes the specified buffer memory and prepares it for writing values.

As with **SETRANGE** (see section 7.1), a value range is specified for the values to be saved. This value range corresponds exactly to one of the 11 possible measuring ranges of the MC-12.

If no range limit is specified, the buffer receives the coarsest range (4.88V).

With **BUFOPEN**, the currently set **SCANTIME** and **PREHIST** parameters are simultaneously transferred to the buffers. However, this only affects the subsequent graphical display on the plotter or oscilloscope.

Example: **BUFOPEN 5,2.4**

| | | |
|---|---|---|
| Error messages: | **ERROR 19** | Invalid buffer number |
| | **ERROR 110** | Invalid range limit |

**BUFRANGE** buffer_number

Abbreviation: **BUFRA**.

buffer number: 1 ... number of buffers

As a check, this function supplies the range of values stored in the buffer in the same way as **RANGE().**

Example: **BUFRANGE(1)**

| | | |
|---|---|---|
| Error messages: | **ERROR 19** | Invalid buffer number |

**BUFWRITE** buffer_number, position, value

Abbreviation: **BUFW.**

| | |
|---|---|
| buffer number: | 1...number of buffers |
| position: | 1...buffer length |
| value: | -4.92...+4.88 (within the **BUFRANGE**) |

Writes the specified world value to the respective buffer memory at the specified position

The values must lie within the specified value range of the buffer (**BUFRANGE**).

Example: **BUFWRITE 1,155,2.2**

| | | |
|---|---|---|
| Error messages: | **ERROR 19** | Illegal buffer number |
| | **ERROR 121** | Illegal position |
| | **ERROR 109** | Value outside the permissible range |

## 10.3 TRANSFORMING THE BUFFER MEMORY

Even without a BASIC program that accesses the buffer memory with **BUFREAD** and **BUFWRITE**, the stored signals can be transformed.

To do this, simply specify the desired transformation function in the **LOADBUFFER** command.

It must also be ensured that the value range of the function lies within the value range of the buffer (**BUFRANGE**), as specified by **BUFOPEN** (see section 10.3) or during a measurement.

**LOADBUFFER** buffer_number, function

Abbreviation: **LOA.**

buffer number: 1 ... Number of buffers
function: any numeric expression with argument **BUFFER(), POSITION**

The specified buffer memory is loaded with the values of the specified function. Arguments in the function expression are:

**BUFFER** buffer_number
**POSITION**

Abbreviation: **BUFF. POS.**

During the **LOADBUFFER** command, **BUFFER(I)** runs through all values stored in Buffer I in sequence and **POSITION** runs through all position values from 1 to **BUFLEN**.

With **LOADBUFFER** it must be strictly ensured that the calculated function values lie within the value range of the buffer.

At the same time, the values should exhaust this range as they are normally displayed as 8-bit values. Otherwise, it can happen that the values within the 8-bit resolution all result in 0.

The function values are calculated within the **BUFLOAD** command using the normal BASIC functions. As, for example, 1024 values have to be calculated with a buffer length of 4 blocks,

the execution time of the **BUFLOAD** command can increase considerably depending on the function used.

**BUFLOAD** can be canceled with the <BREAK> button.

After **BUFLOAD** is aborted by <BREAK> or an error, **POSITION** contains the buffer position currently being processed, **BUFFER()** returns the values of the buffers at this position.

| | | |
|---|---|---|
| Error messages: | **ERROR 19** | Illegal buffer number |
| | **ERROR 109** | Value outside the permissible range |

Examples:

```
LOAD BUFFER 1,0
```
Buffer 1 is deleted

```
LOADBUFFER 1,BUFFER(2)
```
Buffer 2 is copied to buffer 1

```
LOADBUFFER 1,ABS(BUFFER(1))
```
Amount calculation (detector)

```
A=BUFRANGE(3)/(BUFRANGE(1)+BUFRANGE(2))
LOADBUFFER 3,A*E (BUFFER(1)+BUFFER(2))
```
the sum of buffer 1 and buffer 2 is stored in buffer 3

```
A=BUFRANG(1)
B=2*PI/BUFLEN
RADIAN
LOADBUFFER 1, A*SIN(B*POSITION)
```
Buffer 1 is loaded with a sine period

## 10.4 EXAMPLES OF ACCESS TO BUFFER

**Example program: Printing the values of a buffer**

The following program prints the values stored in buffer 1 on the printer:

```
10 LPRINT " BUFFER 1:"
20 LPRINT " POS:";TAB7;" VALUE:"
30 FOR I=1 TO BUFLEN
40 BUFREAD 1,1,A
50 LPRINT I;TAB 7;A
60 NEXT I
```

**Example program: Simple logger**

The following program saves the current measured value at channel 1 in buffer 1 every full minute (see section 7.4)

If the measurement is made with 11 bits, **BUFINIT** can be written in line 10 instead of **DBUFINIT**.

| Program: | Note: |
|---|---|
| `10 BUFINIT 1` | |
| `20 SETRANGE 1,4.8` | same area for input |
| `30 BUFOPEN 1,4.8` | and buffer |
| `40 AUTORANGE OFF` | |
| `50 FOR I=1 TO BUFLEN` | |
| `60 T=TIME` | |
| `70 IF INT(T*100)/100<>T GOTO 60` | Wait for full minutes |
| `80 BUFWRITE 1,I,CHA(1)` | Save measured value |
| `90 SLEEP 58` | Delay (idle state) |
| `100 NEXT I` | |
| `110 MCOFF` | |

**Example program: Integral function**

The following program first fills Buffer 1 with a sine period, then the integral function is calculated and stored in buffer 2.

Finally, both buffers are displayed on the plotter.

| Program: | Notes: |
|---|---|
| `10 BUFINIT 2.1` | 2 buffers with 256 values |
| `20 A=4:RADIAN` | |
| `30 B=2*PI/BUFLEN` | |
| `40 LOADBUFFER 1,A*SIN(B*POSITION)` | Sine -> Buffer 1 |
| `50 BUFOPEN 2` | |
| `60 DX=2*PI/256` | Delta X |
| `70 S=-4` | Initial condition |
| `80 FOR I=1 TO BUFLEN` | |
| `90 BUFREAD 1,1,F` | |
| `100 S=S+F*DX` | Integration of |
| `110 NEXT 1` | |
| `120 PLOT 1,256,1;1,2` | Plotting of both buffers (see section 11.3) |

# Chapter 11

**GRAPHICAL REPRESENTATION OF THE MEASURED VALUES**

The signal curves stored in the buffers can be displayed graphically on an oscilloscope connected to the MC-12 or the CE-150 plotter.

## 11.1 DISPLAY ON THE OSCILLOSCOPE

Any oscilloscope can be used as a peripheral device for the MC-12.

The oscilloscope is connected to the analog outputs of the MC-12 as described in section 2.6.

With a dual-channel oscilloscope, two signal curves stored in the buffers can be displayed simultaneously on the oscilloscope screen. The resolution is 256*2 56 points per signal.

Since the two analog outputs of the MC-12 generate the oscilloscope image, they can no longer be used for control purposes during oscilloscope operation.

**OUTSCREEN** <Channel_number>, <Buffer_number>, <Position>, <Number_of_blocks>

Abbreviation: **OUTS**.

| | |
|---|---|
| Channel_number: | 1 or 2 |
| Buffer_number: | 1 ... number of buffers |
| Position: | 1 ... buffer length |
| Number_of_blocks: | 1 ... buffer length/256 |

This command displays the signal stored in the specified buffer on the oscilloscope. The trace number indicates which analog output of the MC-12(A) the image signal appears at.

Starting at the respective position, the specified number of measured values (in blocks) is displayed. If the number of blocks is N>1, the values are displayed in a compressed format, with only every nth value being displayed.

The number of blocks can also be selected as 1/2, 1/4, 1/8, 1/16. In this case, the values are stretched, and the intermediate values are interpolated.

Example: **`OUTSCREEN 2,1,256,1`**

| | | |
|---|---|---|
| Error messages: | **ERROR 19** | Illegal beam or buffer no. |
| | **ERROR 101** | MC-12 switched off |
| | **ERROR 121** | Illegal position |
| | **ERROR 135** | Battery voltage too low |

**SCREEN ON**
**SCREEN OFF**

Abbreviation: **SCR.O.   SCR.OF.**

These commands are used to switch the signal display on the oscilloscope on or off.

When the image is switched on, the operating speed of the PC-1500 is only about half of the normal operating speed.

Initialization: Screen switched off

| | | |
|---|---|---|
| Error messages: | **ERROR 101** | MC-12 switched off |
| | **ERROR 135** | Battery voltage too low |

**SCREEN**

Abbreviation: **SCR.**

Takes you to the CMOS screen menu, in which the signals stored in the buffers can be examined:

A pulsating point of light appears on the oscilloscope head screen, which acts as a cursor.

The PC-1500 display shows the number of the respective buffer memory, the exact voltage value at the cursor position and the time from the trigger point to the cursor position. and the time from the trigger point to the cursor position.

The cursor can be moved by pressing individual buttons on the oscilloscope screen. For longer signals, the screen is moved along the signal as a viewing window (scrolling).

The image can be stretched or compressed. When stretching, the image is automatically interpolated so that the waveform of the signal becomes more visible.

A copy of the screen image can be made on the plotter at the touch of a button.

**The following button commands can be issued in this operating mode:**

| **Button:** | **Function:** |
|---|---|
| <→> | Moves the cursor to the right; if the cursor reaches the edge position, the viewing window is moved to the right (scrolling). |
| <←> | Moves the cursor to the left. |
| <◆> | Changes the beam on which the cursor is flashing. |
| <SHIFT> | Switches the synchronous operation of both beams on/off. In synchronous mode, all operations for both beams are carried out simultaneously. |
| <↑> | Brings the next buffer memory to the display. |
| <↓> | Displays the previous buffer memory. |
| <*> | Stretches the display (interpolation) |
| </> | Compresses the display. |
| <N> | Normal display (1:1) |
| <+> | Increases the cursor step size (3 levels) |
| <-> | Reduces the cursor step size (3 levels) |
| <H> | Hardcopy of the curve under the cursor on the CE-150 plotter. In SHIFT mode, both curves are drawn. |
| <E> | End, return to BASIC |
| <BREAK> | Abort |

## 11.2 DISPLAY ON THE CE-150 PLOTTER

When displaying the stored signals on the plotter, the curves are provided with a labeled coordinate system.

The label of the Y-axis is selected based on the stored measuring range (**BUFRANGE**). The label of the time axis is calculated from the stored history and sampling time. Up to 5 curves can be drawn in one image.

**PLOT** <startpos>, <endpos>, <compression>, <bufferno>, <bufferno>, ...

Abbreviation: **PL.**

| | |
|---|---|
| Startposition: | 1 ... buffer length |
| Endposition: | startpos ... buffer length |
| Compression: | 0 ... buffer length/256 |
| Buffer no: | 1 ... buffer count |

Graphically displays the signal curve stored in the specified buffers between the start position and end position on the CE-150 plotter.

Multiple buffers are drawn in different colors in the same coordinate system. A maximum of five buffers can be combined in one representation.

If multiple buffers are drawn in an image, they must have the same time axis (same **SCANTIME** and **PREHIST**).

With a compression factor of 1, the image is displayed 1:1, with each pixel corresponding to one measured value.

Analogous to **OUTSCREEN**, the image can be compressed (compression_factor 2, 3, ...) or stretched (compression_factor 1/2, 1/4, 1/8, 1/16).

The maximum image length is 768 pixels. If the entered range exceeds this maximum length, the excess values are not drawn. Large buffers whose representation should be output 1:1 must therefore be drawn in multiple sections.

Example: **PLOT 1,512,1;1,2**

| | | |
|---|---|---|
| Error messages: | **ERROR 121** | Invalid position specification |
| | **ERROR 19** | Invalid buffer number |
| | **ERROR 123** | Buffer with different time axis |

# Appendix A

**LIST OF ERROR MESSAGES**

| | |
|---|---|
| 19. | The value of the numeric expression is outside the allowed range.<br><br>Example: **CHA 6** |
| 101. | To execute the instruction, the analog part of the MC-12 must be switched on with the **MCON** command.<br><br>Example: **MCOFF**<br>**CHA 1** |
| 102. | An attempt was made to load the program for transient recorder operation even though a BASIC program is present in the PC-1500's memory. Delete the BASIC program with **NEW.** |
| 109. | The trigger level is outside the set measurement range. Or: While writing to a buffer, the value to be written is outside the range -4.88...+4.88.<br><br>Example: **SET RANGE 1.0.5**<br>**SETTRIGGER 4**<br>**SELECT +1.1**<br>**INSCAN** |
| 110. | Overflow<br>The measured value exceeds the set range limit. |
| 111. | The scan time set with **SETSCANTIME** is too short.<br><br>Example: **SETSCANTIM E50E-6**<br>**SELECT +1 ,1; 2, 2**<br>**INSCAN** |
| 121. | When accessing a buffer, the position value is outside the permissible range. Only values from 1 to **BUFLEN** are possible for the position.<br><br>Example: **BUFREAD 1,0,A** |

| | |
|---|---|
| 123. | Buffer memories with different time bases or histories were specified in the **PLOT** command. Only those signal waveforms measured with the same sampling interval and the same histories can be displayed in a single image. |
| 125. | The buffer memory must be prepared with the **BUFOPEN** command before being written to with **BUFWRITE**.<br><br>Example: **BUFWRITE 1,100,4.02** |
| 130. | More than five channels were specified in the **SELECT** command.<br><br>Example: **SELECT +1,1;2,2;3,3;4,4;5,5;6,6** |
| 131. | In the **SELECT** command, the signal waveforms of two input channels were assigned to the same buffer. A separate buffer must be reserved for each channel to be measured.<br><br>Example: **SELECT +1,1;2,1** |
| 135. | The voltage of the batteries in the CE-150 is no longer sufficient to power the MC-12. Connect the AC adapter to the CE-150. |

# Appendix B

## LIST OF FUNCTIONS, CONTROL VARIABLES AND INSTRUCTIONS

### Functions

| Function | Abbreviation | Remarks | Page |
|---|---|---|---|
| **CHA** | | Current measured value at the specified input channel | 49 |

### Control Variables

| Control Variable | Abbreviation | Remarks | Page |
|---|---|---|---|
| **BUFFER** | **BUFF.** | Buffer value at the current **POSITION** in the specified buffer (in conjunction with **LOADBUFFER**) | 70 |
| **BUFLEN** | **BUFL.** | Buffer length in measured values (set by **BUFINIT**) | 60 |
| **BUFNUM** | **BUFN.** | Buffer count (set by **BUFINIT**) | 60 |
| **POSITION** | **POS.** | Position within the buffer (in conjunction with **LOADBUFFER**) | 70 |
| **PREHIST** | **PRE.** | Number of history blocks (in conjunction with **SETPREHIST**) | 71 |

| | | | |
|---|---|---|---|
| **RANGE** | **RANG.** | Range limit at the specified input channel<br>(in conjunction with **SETRANGE**) | 48 |
| **SCANTIME** | **SC.** | Sampling time<br>(in conjunction with **SETSCANTIME**) | 63 |
| **TRIGGER** | **TRI.** | Trigger level in volts<br>(in conjunction with **SETTRIGGER**) | 61 |

**Commands**

| Command | Abbreviation | Remarks | Page |
|---|---|---|---|
| **AUTORANGE OFF** | **AU. OF.** | Turns on automatic range selection. | 49 |
| **AUTORANGE ON** | **AU. O.** | Turns off automatic range selection. | 49 |
| **BUFINIT** | **BUFI.** | Sets the number of buffers. | 58 |
| **BUFOPEN** | **BUFO.** | Clears the specified write buffer. | 68 |
| **BUFREAD** | **BUFR.** | Assigning a buffer value to a variable | 68 |
| **BUFWRITE** | **BUFW.** | Writes a value at the specified position to the specified buffer. | 69 |
| **INCHA** | **INC.** | Assigning the current measured value to a variable | 50 |
| **INFUNCTION** | **INF.** | Assigning the current, transformed measured value to a variable. The transformation function is specified using the SETFUNCTION command. | 51 |
| **INIT B** | **INI. B** | Initializes the measuring system for bipolar mode. | 44 |
| **INIT U** | **INI. U** | Initializes the measuring system for unipolar mode. | 44 |

| | | | |
|---|---|---|---|
| **INIT** | **INI.** | Initializes the measuring system without changing the operating mode. | 44 |
| **INSCAN** | **INS.** | Starts an indirect measurement. | 64 |
| **LOADBUFFER** | **LOA.** | Writes the entire specified buffer with the specified function. | 70 |
| **MCOFF** | **MCOF.** | Turns off the analog part of the HC-12. | 44 |
| **MCON** | **MCO.** | Turns on the analog part of the HC-12. | 44 |
| **MULTIMETER** | **MU.** | Starts the MULTIMETER dialog program | 31 |
| **OUTCHA** | **OU.** | Applies the specified voltage to the specified output channel. | 49 |
| **OUTSCREEN** | **OUTS.** | Outputs measured values stored in a buffer to a connected oscilloscope. | 73 |
| **PLOT** | **PL.** | Outputs measured values stored in one or more buffers to the CE-150 plotter. | 76 |
| **RELAY OFF** | **REL. OF.** | Turns off the specified remote switch on the CE-150. | 56 |
| **RELAY ON** | **REL. O.** | Turns on the specified remote switch on the CE-150. | 56 |
| **SCREEN** | **SCR.** | Calls up the CMOS screen menu (only with an oscilloscope connected). | 74 |

| | | | |
|---|---|---|---|
| **SCREEN OFF** | **SCR. OF.** | Turns off the screen display (only with an oscilloscope connected). | 74 |
| **SCREEN ON** | **SCR. O.** | Turns on the screen display (only with an oscilloscope connected). | 74 |
| **SELECT** | **SE.** | Selects input channels for indirect measurements and assigns buffer memory to them. | 63 |
| **SETFUNCTION** | **SETF.** | Defines a function that is used to transform the current measured value in the INFUNCTION command. | 50 |
| **SETPREHIST** | **SETP.** | Specifies the number of history blocks to capture. | 60 |
| **SETRANGE** | **SET.** | Sets the measuring range limit on the specified input channel. | 48 |
| **SETSCANTIME** | **SETS.** | Specifies the sampling time for indirect measurements. | 62 |
| **SETTRIGGER** | **SETT.** | Sets the trigger level for indirect measurements. | 61 |
| **SWITCH OFF** | **SW. OF.** | Turns off the specified analog switch. | 55 |
| **SWITCH ON** | **SW. O.** | Turns on the specified analog switch. | 55 |
| **TRANSREC** | **TRA.** | Loads the BASIC program TRANSIENTRECORDER into the program memory of the PC-1500. | 32 |

# MC-12A

# Additional description

# Table of contents

## MC-12A Supplementary Description

# Chapter 1:

# V24 Interface

## 1.1 Functional description

The MC-12A features an interface commonly known as V.24 or RS232C. This is an interface for serial data transmission in asynchronous mode.

The MC-12A behaves like a terminal (send and receive) and operates all control lines except the CD line (PIN 8). The interface's transmission speed can be set in 6 steps from 300 baud to 9600 baud. The speed levels 4800 baud and 9600 baud require the full functionality of control lines 4 and 5 on the connected receiver or transmitter.

The connection of devices to the interface is shown in Figure 1.

## 1.2 Interface control commands

**`COM ON`**
**`COM OFF`**

These commands turn the interface on and off. This command is independent of the **`MC ON`** and **`MC OFF`** commands.

When the PC 1500(A) is switched off, the interface is also switched off.

**SETCOM** Baud, Stopbits, Parity, LF

| | |
|---|---|
| Baud: | 1= 300<br>2=600<br>3= 1200<br>4=2400<br>5= 4800<br>6= 9600 |
| Stop bits: | 1 = one stop bit, 2 = two stop bits |
| Parity: | 0 = NO; 1 = ODD; 2 = EVEN |
| LF: | 0= only CR; 1= automatically LF after CR |

This command sets the interface parameters. The number of data bits is always 7. A parity bit is sent after the data bit. If **NO** Parity is set, the level is at logic 0. Two stop bits are sent after the parity bit.

With LF, if desired, a line feed can be sent automatically after each carriage return or the following line feed can be ignored when receiving after each carriage return.

**COM LIST** line no. 1, line no. 2

This command outputs the currently active BASIC program from line 1 to line 2 to the RS-232 interface. If no line numbers are specified, the entire program is listed.

Using this command, PC-1500(A) programs can be transferred to another computer (e.g. for storage on a floppy disk).

**COM INPUT** n

This command only works with the PC 1500(A) and switches input from the keyboard to the RS-232 interface until the character with the ASCII code 'n' is received or the BREAK key is pressed.

If the PC-1500(A) is in PRO mode when this command is entered, program files can be transferred from another computer to the PC-1500(A). Typically (CP/M, MS-DOS), CTRL-Z with the code value &1A is used as the termination character.

**PRINT#-232**, (numbers or strings)

This command sends data over the V.24 interface. The command functions exactly the same as the LPRINT command. USING statements are handled as usual.

**INPUT#-232**, (numbers or strings)

The INPUT command allows the reception of data via the V.24 interface

## 1.3 Example program for data transfer

The following program demonstrates the transfer of measurement data from the MC-12A to another computer with a V.24 interface.

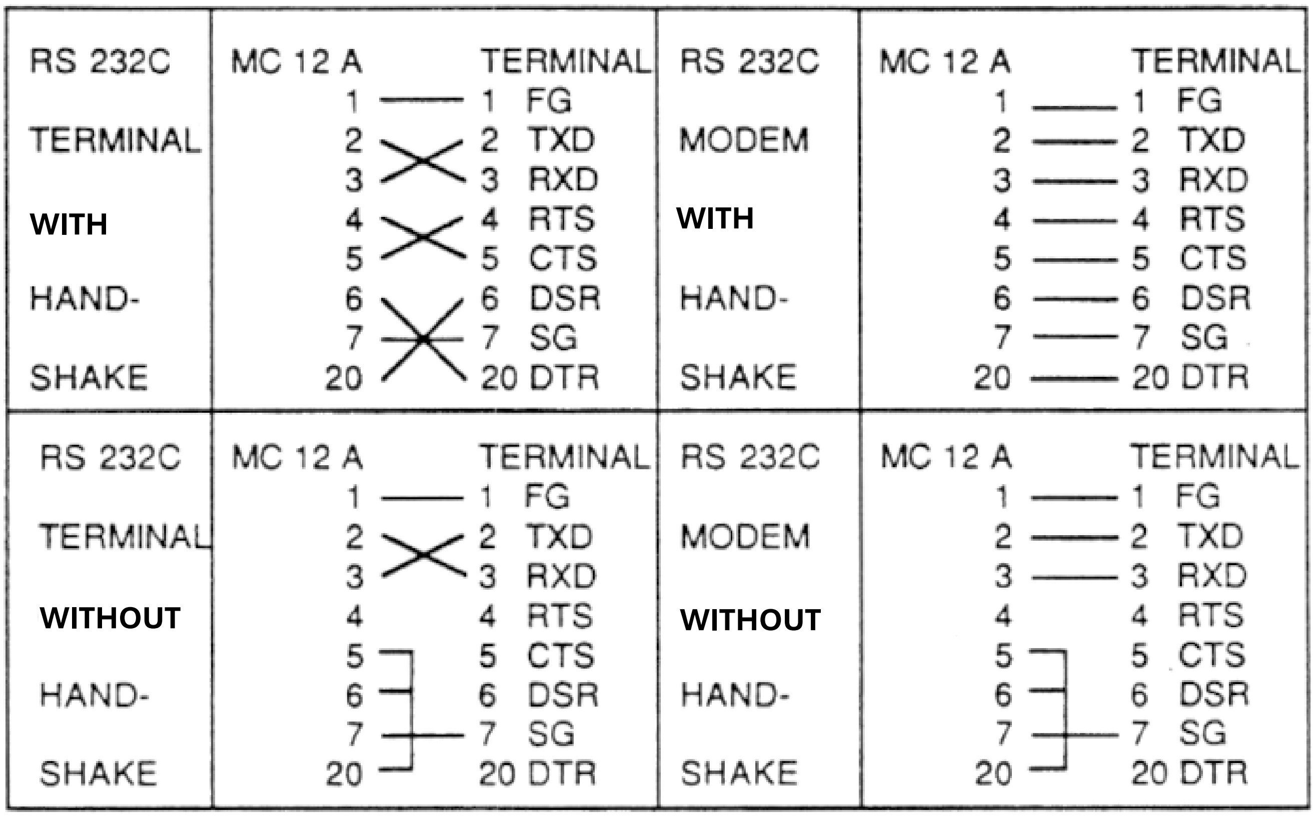
The command SETCOM 4,0,0 sets a transmission rate of 2400 baud. No parity is generated, and no LF is sent after CR.

Example program:

```
10 SETCOM 4,0,0 : COM ON
20 FOR I=1 TO BUFLEN
30 BUFREAD 1,1,A
40 PRINT#-232,A
50 NEXT I
60 COM OFF : END
```

## 1.4 CONNECTION OF THE V.24 INTERFACE

The following diagram shows the different connection options for the V.24 interface, depending on whether the device to be controlled is a terminal or a modem, with or without handshake lines.




For some devices, PIN 8 of the connector must also be connected to PIN 20 to enable operation. Further information can be found in the descriptions of the devices to be connected.

# Chapter 2:

# I/O Port

## 2.1 Description of the I/O

The MC-12 A is equipped with a digital input/output module with the type designation 8255. This module is fully available to the user and opens up a wide range of applications. The possibilities range from communication with other data processing systems to the control and monitoring of the most complex systems.

The 24 digital control lines can be defined as inputs or outputs in groups of 2x8 and 2x4 lines. Programming can be done in both BASIC and machine language.

## 2.2 Pin assignment and programming

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Vbat | PB3 | PB2 | PB1 | PB0 | PC2 | PC0 | PC5 | PC7 | PA0 | PA1 | PA2 | PA3 |
| 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| GND | PB4 | PB5 | PB6 | PB7 | PC3 | PC1 | PC4 | PC6 | PC7 | PA6 | PA5 | PA4 |

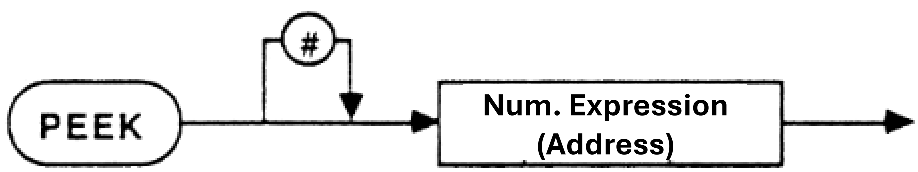
**below**

The peripheral interface module 8255 is addressed via the addresses 2000H - 2003H (hexadecimal) on the 2nd memory page of the PC 1500(A).

The control of the 3 8-bit output channels and the control logic is shown in the following table.

| Address | Function | Access |
|---|---|---|
| 2000H | Channel A data bus | Reading |
| 2001H | Channel B data bus | Reading |
| 2002H | Channel C data bus | Reading |
| 2003H | Control logic data bus | Reading |
| 2000H | Data bus channel A | Writing |
| 2001H | Data bus channel B | Writing |
| 2002H | Data bus channel C | Writing |
| 2003H | Data bus control logic | Writing |

A detailed description of the 8255 follows in Chapter 2.2. The most important functions will be illustrated with examples below. Programming is done in BASIC. The POKE and PEEK commands are required. These commands are not described in the PC 1500 (A) manual and are briefly introduced here.




Returns the decimal contents of the memory location specified by the numeric expression. PEEK refers to the first memory block, PEEK# to the second memory block.

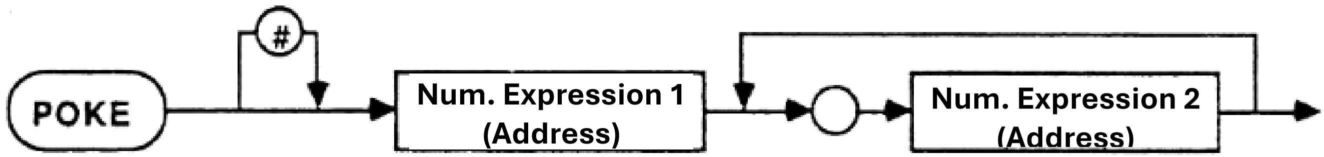
Example:

**PEEK# &2000**

Reads the information from channel A of the 8255

&2000 = 8192: The leading & sign indicates that the address is hexadecimal.

Value range:
Addresses - 0...65535; 0...&FFFF




Stores the data specified by numeric expression 2 at the address specified by numeric expression 1 and its subsequent addresses. POKE refers to the first memory block, POKE# to the second memory block.

Example:

```
POKE# &2000,1,254,2
```

writes data 1, 254, and 2 to channels A, B, and C of the 8255 port IC

Value range:
Addresses - 0...65535; 0...&FFFF
Data - 0...255; 0...&FF

The number of data bytes that can be entered by a single POKE command is limited only by the maximum line length.

## 2.3 Example program

The following program defines all 24 lines of the Port IC as inputs and represents the logical state of each line as 0 or 1.

In line 10 of the program, the I/O port is initialized in operating mode 0 and all lines are switched as inputs.

In line 20, the states at the 3 ports A, B and C are loaded into the variables A, B and C.

In the subsequent program loop, each individual bit of the variable is checked and the result is displayed as 0 or 1.

```
10 "A":POKE# &2003,155 : WAIT 0
20 CURSOR 0 : A=PEEK# &2000 : B=-PEEK# &2001 : C=-PEEK# &2002
30 FOR J=1 TO 3 : FOR I=7 TO 0 STEP -1
40 D=2^I
50 IF @(I) AND D PRINT "1"; : NEXT I
60 PRINT ” "; : NEXT J
70 GOTO 20
80 END
```

OUTPUT:

| PA7...........PA0 | PB7...........PB0 | PC7...........PC0 |
|---|---|---|
| 10010001 | 00010001 | 01000100 |
| PORT A | PORT B | PORT C |

0 = logischer Pegel = 0 (=.0 V)
1 = logischer Pegel = 1 (< 2.2 V)

# Programmable peripheral interface module:  SAB 8255

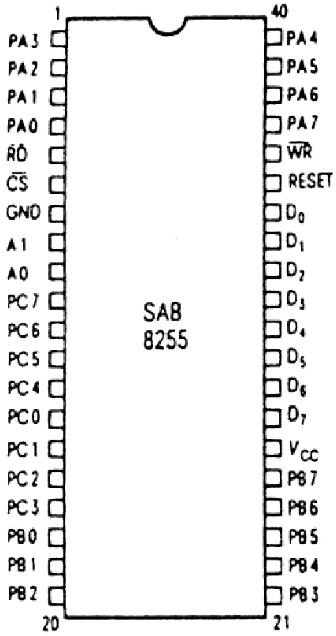
**24 programmable I/O ports**
**Fully TTL compatible**
**Fully compatible with the SAB 8080 microprocessor family**
**Direct bit set and reset capabilities to simplify interfacing in control applications**
**40-pin DIL package**

The SAB 8255 is a programmable general-purpose I/O module for the SAB 8080 microprocessor. It has 24 I/O ports, which can be programmed separately in two groups of two ports each and used in three main modes. In the first mode (Mode 0), each group of 12 I/O ports can be programmed as input or output in sections of four ports.

In the second operating mode (Mode 1), eight lines of each group can be programmed as inputs or outputs. Of the remaining four connections, three are used for handshake and interrupt control signals. The third operating mode (Mode 2) can be described as two-way bus mode, in which eight connections are used for a two-way bus. Five additional connections, one of which belongs to another group, are used for handshake in this case.

In addition, the direct setting and resetting of individual bits is possible. The maximum current is 1 mA at a voltage of 1.5 V. This allows Darling Gate transistors to be directly controlled in applications for printers and high-voltage display units.

## Pin assignment

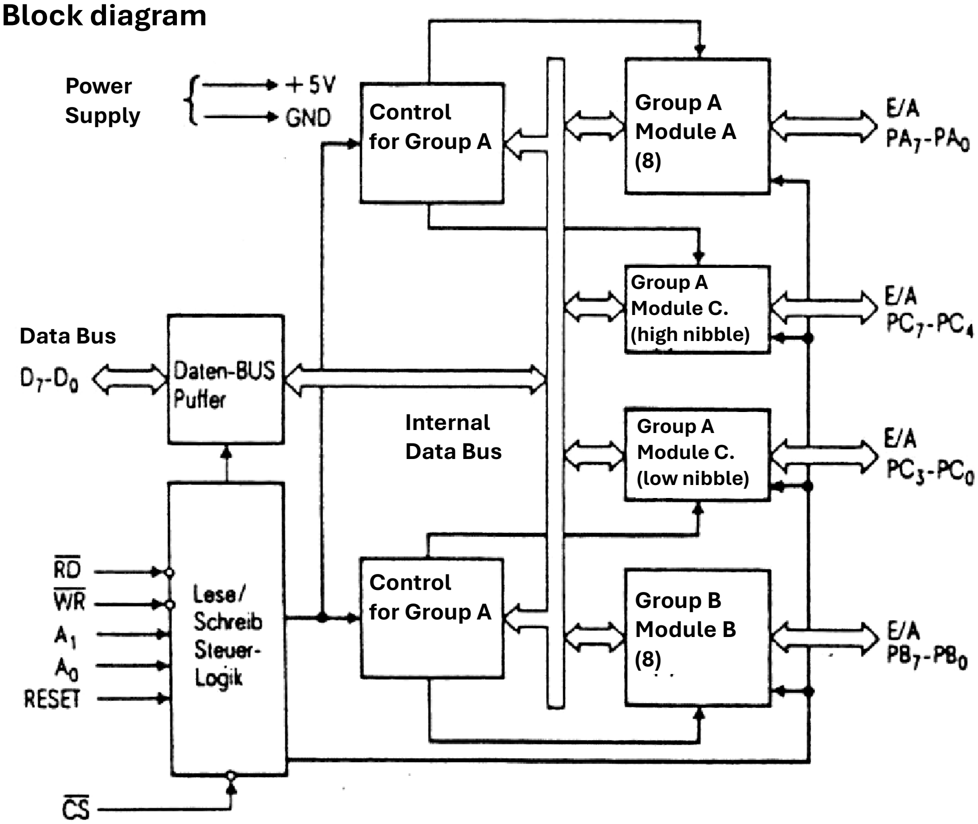



## Connection designations

| | |
|---|---|
| $D_0$–$D_7$ | Data bus (2 way) |
| RESET | Reset Input |
| $\overline{CS}$ | Building block selection |
| $\overline{RD}$ | Read input |
| $\overline{WR}$ | Write input |
| A0, A1 | Channel address |
| $PA_0$–$PA_7$ | Channel A (Bit 0 to 7) |
| $PB_0$–$PB_7$ | Channel B (Bit 0 to 7) |
| $PC_0$–$PC_7$ | Channel C (Bit 0 to 7) |
| $V_{cc}$ | Supply Voltage (+5V) |
| GND | Ground (0V) |

Dimensions Page 398

## Block diagram



## Functional description

### General

The SAB 8255 is a programmable peripheral interface (PPS) module for the SAB 8080 microcomputer system. As a general-purpose I/O module, it connects peripheral devices to the system data bus. The functional properties of the SAB 8255 are determined by software, so no additional logic modules are normally required to connect peripheral devices or circuits.

### Data bus buffer

An 8-bit, two-way buffer with three output states connects the SAB 8255 to the system data bus. Data is output or received from the buffer when the (IN-PUT) and (OUT-PUT) commands are executed. Control values and state information are also transmitted through the data bus buffer.

### Read/write and control logic

This circuit section handles all internal and external transmissions of data, control, or status words. It receives information from the SAB 8080's address and control bus and sends corresponding commands to the control logic of both groups.

**(/CS)**
Chip Select: A LOW level at this input initiates the exchange of information between the SAB 8255 and the SAB 8080.

**(/RD)**
Read: A LOW level at this input allows the SAB 8255 to send data or status information to the SAB 8080 via the data bus.

**(/WR)**
Write: A LOW level at this input allows the SAB 8080 to write data or control words to the SAB 8255.

**(RESET)**
Reset: A HIGH level at this input resets all lower registers, including the control register, and puts all channels (A, B, C) into input mode.

**(A0 and A1)**

Channel Select 0 and Channel Select 1: In conjunction with the /RD and /WR inputs, these input signals control the selection of one of the three channels or the control word register. They are typically connected to the least significant bits (A0 and A1) of the address bus.

| **A1** | **A0** | **/RD** | **/WR** | **/CS** | **Input (Read)** |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | Channel A -> Data BUS |
| 0 | 1 | 0 | 1 | 0 | Channel B -> Data BUS |
| 1 | 0 | 0 | 1 | 0 | Channel C -> Data BUS |
| | | | | | Output (writing) |
| 0 | 0 | 1 | 0 | 0 | Data BUS -> Channel A |
| 0 | 1 | 1 | 0 | 0 | Data BUS -> Channel B |
| 1 | 0 | 1 | 0 | 0 | Data BUS -> Channel C |
| 1 | 1 | 1 | 0 | 0 | Data bus -> control logic |
| | | | | | Functions not selected |
| X | X | X | X | 1 | Data bus -> high-impedance state |
| 1 | 1 | 0 | 1 | 0 | invalid condition |

**Control logic of groups A and B**

The function of each individual channel is programmable via software. This is done by sending a control word to the SAB 8255, which contains information such as "operating mode," "set bit," "reset bit," and other information that determines the functional characteristics of the SAB 8255.

Each of the control blocks (Group A and Group B) accepts "commands" from the read/write and control logic, receives "control words" from the internal data bus and issues the corresponding commands to the corresponding channels.

Control logic, group A — channel A and channel C, high-order bits (C7-C4)
Control logic, group B — channel B and channel C, low-order bits (C3-C0)

The control word register can only be written to. Reading the control word register is not possible.

**Channels A, B and C**

The SAB 8255 contains three 8-bit channels (A, B, and C). They can perform various functions through appropriate software programming. In addition, each channel has special features that further expand the application range and flexibility of the SAB 8255.

**Channel A:** One 8-bit data output latch/buffer and one 8-bit data input latch.

**Channel B:** One 8-bit data input/output latch/buffer and one 8-bit data output latch.

**Channel C:** An 8-bit data output latch/buffer and an 8-bit data input buffer (no input latching). This channel can be split into two 4-bit channels by controlling the operating mode. Each 4-bit channel consists of a 4-bit latch and can be used for control signal outputs in conjunction with channels A and B.

**Detailed operating description**

**Selecting the operating mode**
Three main operating modes can be set by the system software:

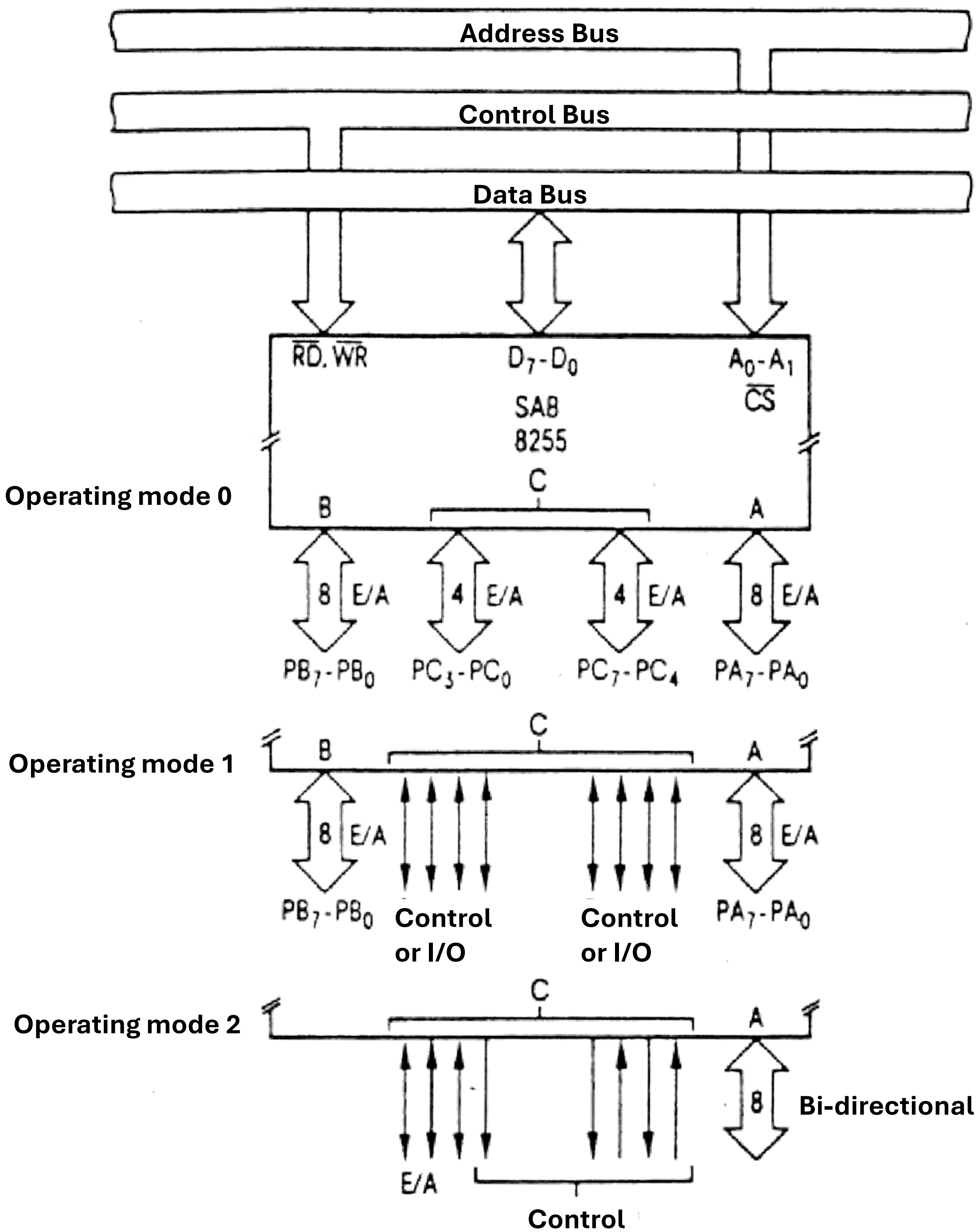
Operating Mode 0: Simple Input/Output
Operating Mode 1: Keyed Input/Output
Operating Mode 2: Two-Way Bus

If the reset input is HIGH, all channels are set to the input state (i.e., the 24 lines have a high input resistance). After the reset signal is removed, the SAB 8255 remains in the input state without any additional settings. Any of the other operating modes can be selected during the execution of a system program (QUT PUT). This allows a single SAB 8255 to operate various peripheral devices with a simple software management program.

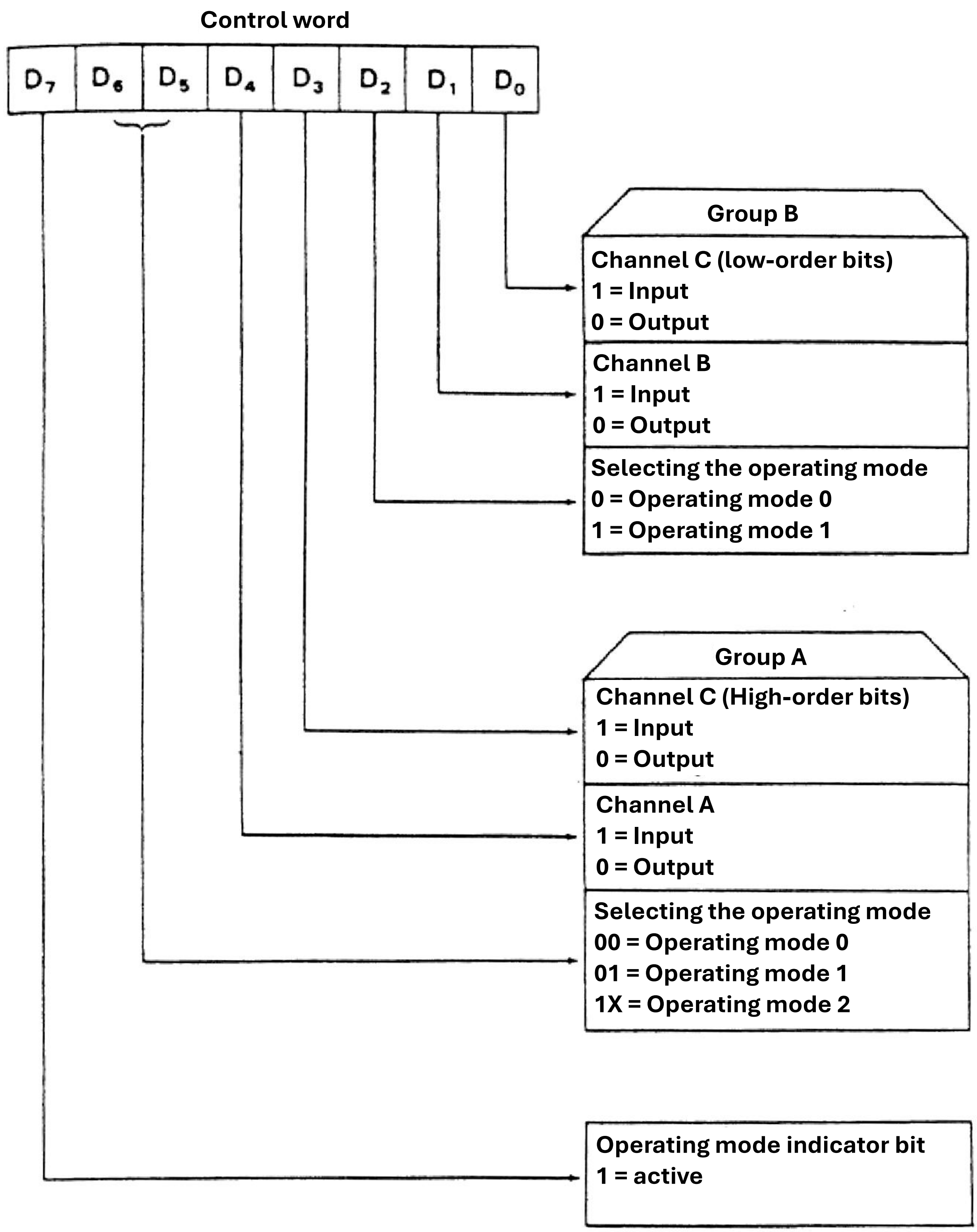
The operating modes of channels A and B can be defined independently, while channel C is split into two parts according to the requirements of channels A and B. When the operating mode is changed, all output registers, including the status flip-flop, are reset. Operating modes can be combined, so their functional definition can be tailored to virtually any I/O structure. For example, group B can be programmed for operating mode 0 to monitor switch closures or display calculation results, while group A could be programmed for operating mode 1 to monitor a keyboard or a punched tape reader through interrupt control.

The possible combinations of operating modes may seem confusing at first glance. However, after a brief overview of the overall functionality of the module, the simple and clear I/O structure becomes apparent.

## Definition of operating modes and the BUS interface



## Format definition for operating mode selection



**Single bit set/reset**
Each of the 8 bits of channel C can be set or reset by an output command (OUTPUT). This feature reduces software complexity in control engineering applications.

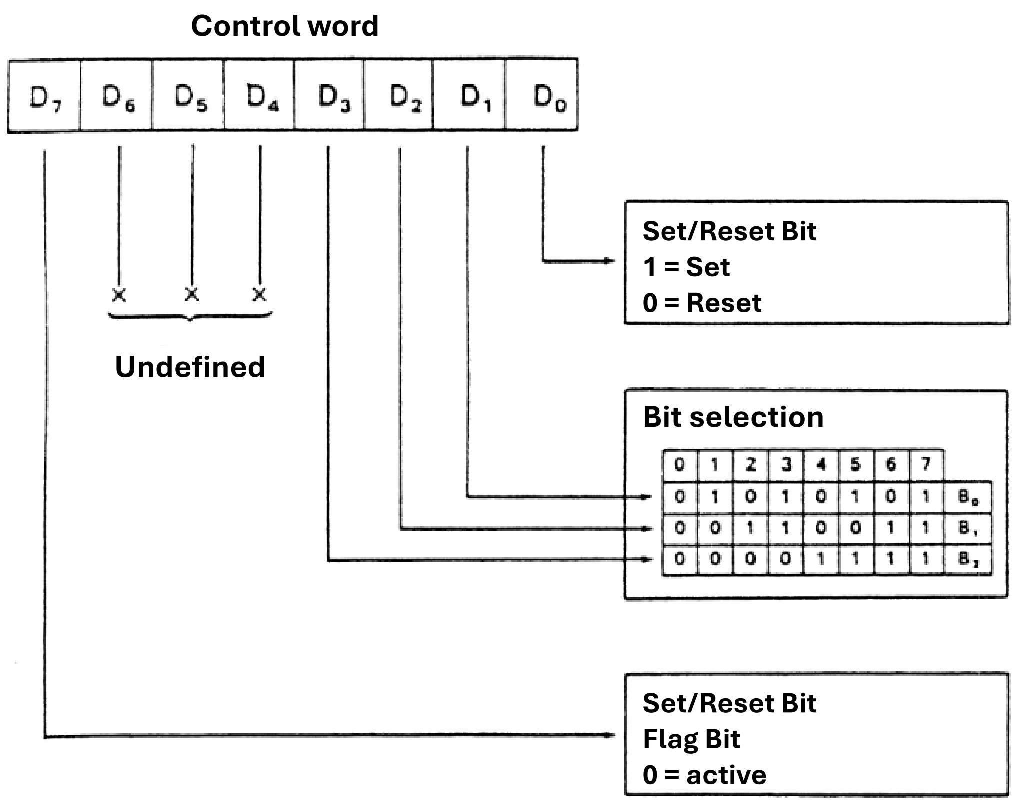
If channel C is used for status and control purposes for channel A or B, the bits can be set or reset using the "Set/Reset Bit" operation, just like a data output channel.

**Interrupt control functions**
If the SAB 8255 is programmed for operating mode 1 or 2, control signals are available that can be used as interrupt request signals for the SAB 8080. The interrupt request signals generated by channel C can be disabled or enabled by setting or resetting the corresponding INTE flip-flop, using the "Bit Set/Reset" function of channel C.

Definitions for the INTE flip-flop:

(Bit-SET) — INTE is set — interrupt enabled
(Bit-RESET) — INTE is reset — interrupt disabled

Note: All masking flip-flops are automatically reset when the operating mode is selected and when the device is reset.




**Format for bit set/reset**

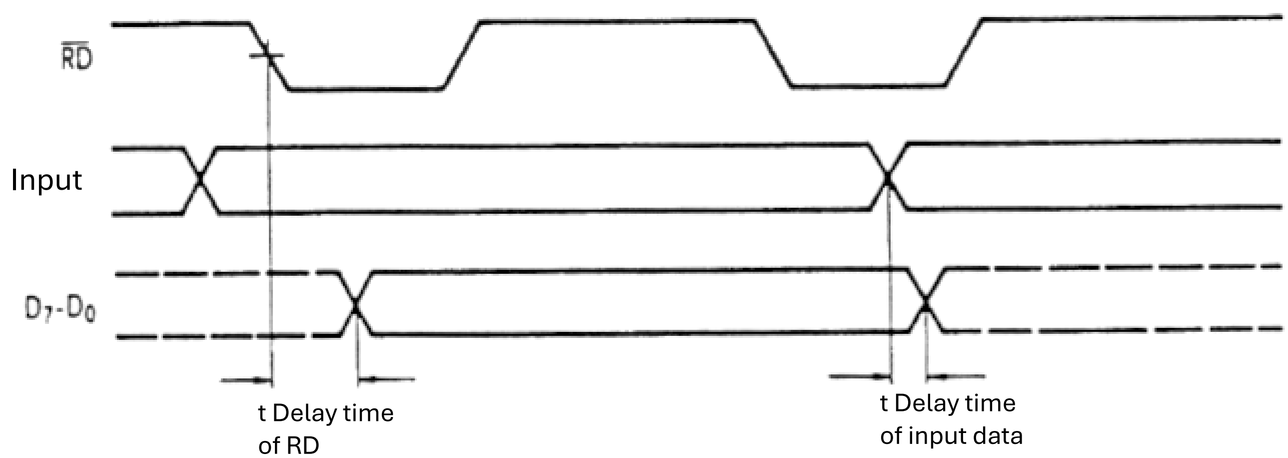
## Operating modes

### Operating Mode 0 (Simple Input/Output)

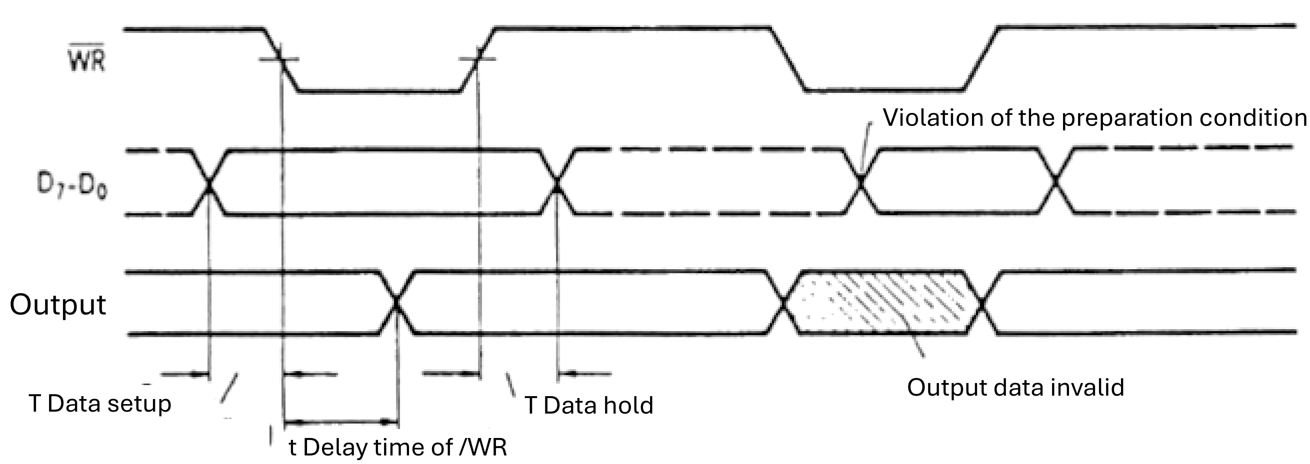
This functional arrangement enables simple input and output for each of the three channels. No handshake is required, as data is simply written to or read from the selected channel.

- Basic functional definitions of operating mode O:
- Two 8-bit channels and two 4-bit channels
- Each channel can be an input or output
- Outputs have buffers
- Inputs work without buffers
- 16 different input/output combinations are possible in this operating mode.

**Basic timing of the input (D7-D0 follow the input, no buffering)**




**Basic timing of the output (outputs with buffering)**



**Definition of the channels for operating mode 0:**

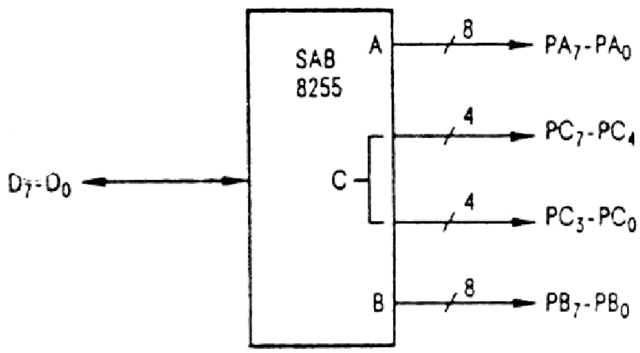
| **A** | | **B** | | **Group A** | | | **Group B** | |
|---|---|---|---|---|---|---|---|---|
| D4 | D3 | D1 | D0 | Channel A | Chanel B (high-order bits) | # | Channel B | Chanel C (low-order bits) |
| 0 | 0 | 0 | 0 | Output | Output | 0 | Output | Output |
| 0 | 0 | 0 | 1 | Output | Output | 1 | Output | Input |
| 0 | 0 | 1 | 0 | Output | Output | 2 | Input | Output |
| 0 | 0 | 1 | 1 | Output | Output | 3 | Input | Input |
| 0 | 1 | 0 | 0 | Output | Input | 4 | Output | Output |
| 0 | 1 | 0 | 1 | Output | Input | 5 | Output | Input |
| 0 | 1 | 1 | 0 | Output | Input | 6 | Input | Output |
| 0 | 1 | 1 | 1 | Output | Input | 7 | Input | Input |
| 1 | 0 | 0 | 0 | Input | Output | 8 | Output | Output |
| 1 | 0 | 0 | 1 | Input | Output | 9 | Output | Input |
| 1 | 0 | 1 | 0 | Input | Output | 10 | Input | Output |
| 1 | 0 | 1 | 1 | Input | Output | 11 | Input | Input |
| 1 | 1 | 0 | 0 | Input | Input | 12 | Output | Output |
| 1 | 1 | 0 | 1 | Input | Input | 13 | Output | Input |
| 1 | 1 | 1 | 0 | Input | Input | 14 | Input | Output |
| 1 | 1 | 1 | 1 | Input | Input | 15 | Input | Input |

## Arrangements in operating mode 0:

**Arrangements in operating mode 0:**

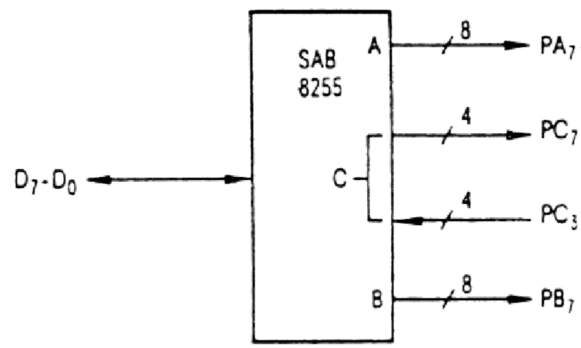
**Control word no. 0**

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |



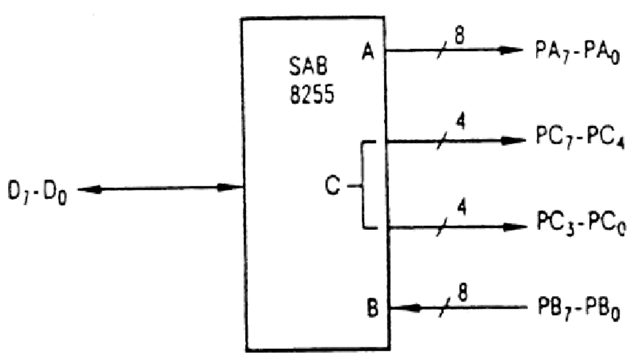

**Control word no. 1**

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |



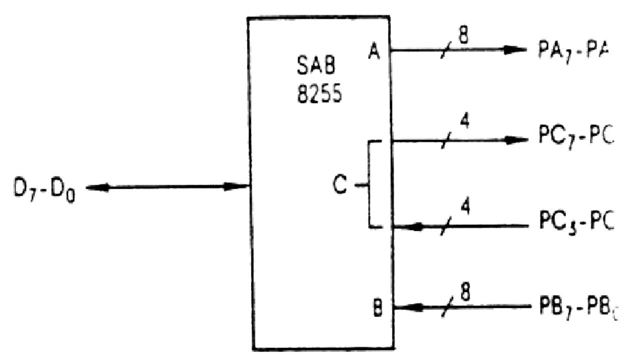

**Control word no. 2**

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |



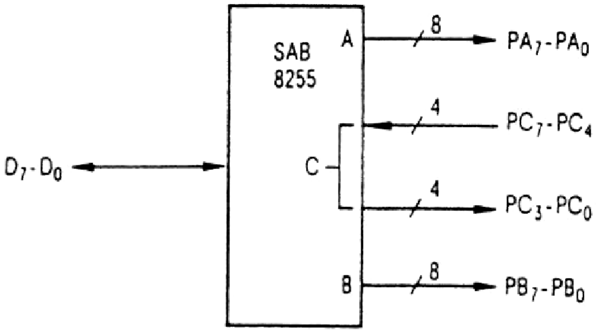

**Control word no. 3**

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |



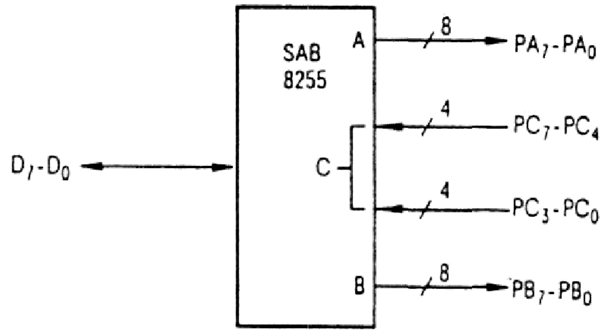
### Control word no. 4

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |



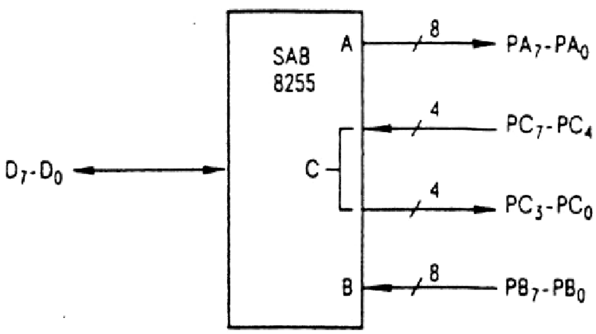

### Control word no. 5

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |



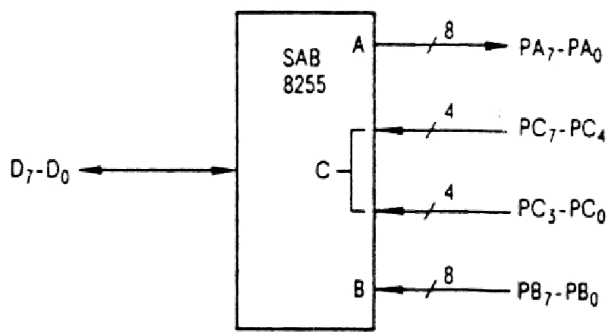

### Control word no. 6

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

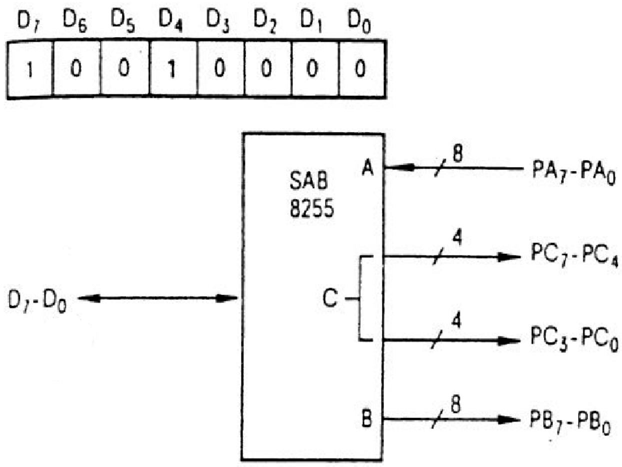



### Control word no. 7

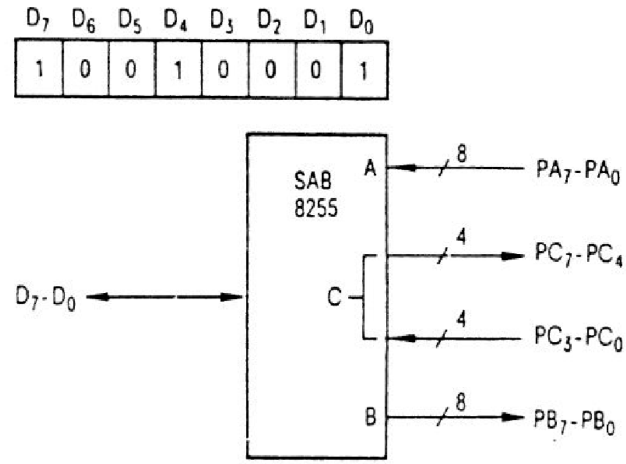
| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |



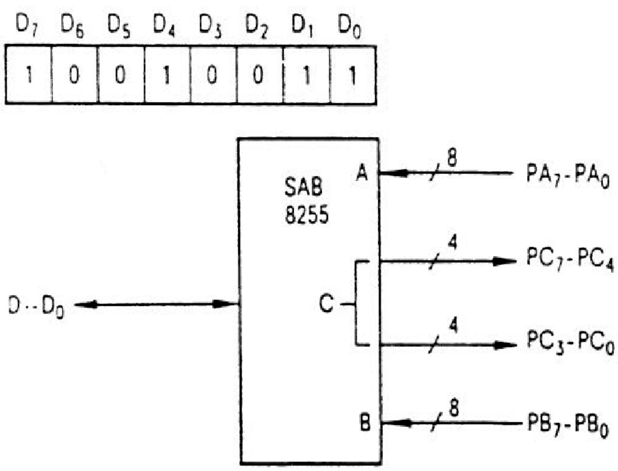
**Control word no. 8**



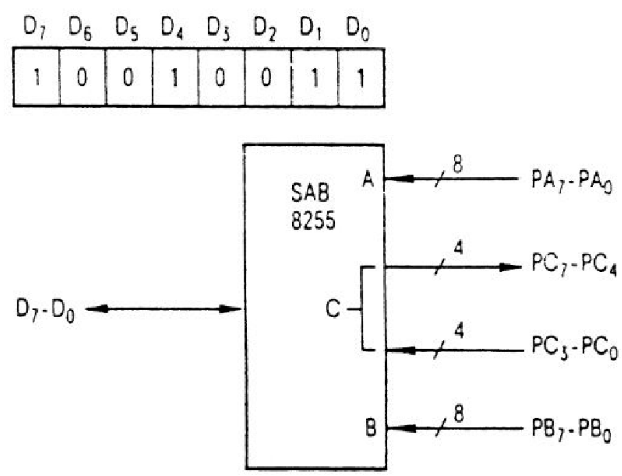

**Control word no. 9**




**Control word no. 10**

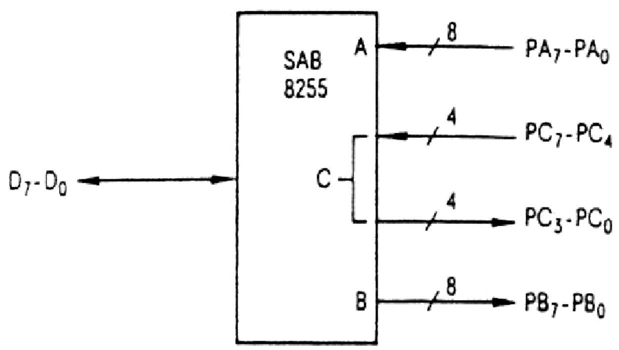



**Control word no. 11**



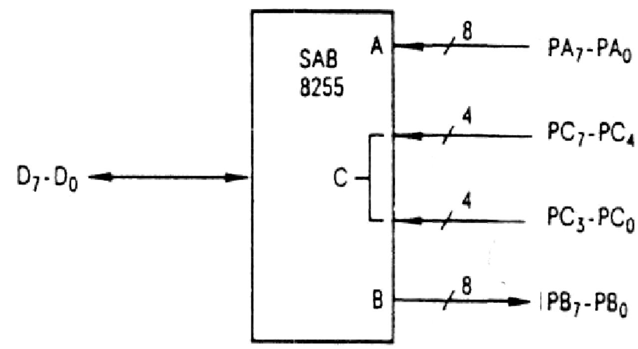
## Control word no. 12

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |



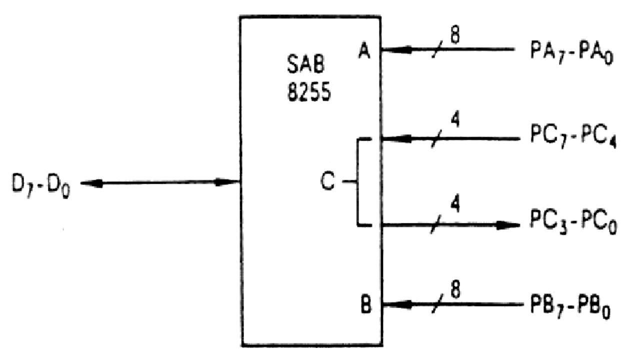

## Control word no. 13

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |



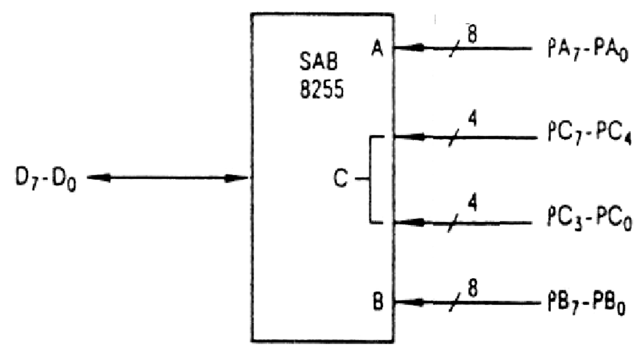

## Control word no. 14

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |




## Control word no. 15

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |



## Operating mode 7 (keyed input/output)

This functional arrangement is used to exchange I/O data to or from a selected channel in conjunction with sampling pulses or "acknowledge" signals. Channel A and Channel B use the lines of Channel C in Mode 1 to generate or receive these "acknowledge" signals.

Basic functional definitions of operating mode 1:

- Two groups (Group A and Group B)
- Each group includes an 8-bit data channel and a 4-bit control/data channel.
- The 8-bit data channel can be used either as an input or output.
- The 4-bit channel is used for control and status purposes for the 8-bit data channels.

**Definition of the input control signals**

**/STB (Save Signal Input)**
A LOW level at this input causes data to be loaded into the input buffer.

**IBF (Input Buffer Flip-Flop, Loaded)**
A HIGH level at this output indicates that data has been loaded into the input buffer; this corresponds to an acknowledgement. IBF is set by the falling edge of the STB input and reset by the rising edge of the RD input.

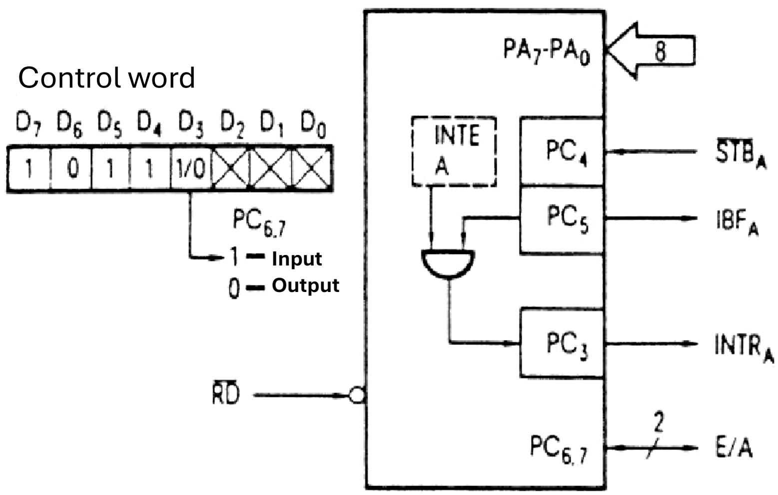
**INTR (Interrupt Request)**
A HIGH level at this output can be used to interrupt the microprocessor's main program when an input device needs to be controlled. INTR is set by the rising edge of STB if IBF is set to "one" and INTE is set to "one." It is reset by the falling edge of RD. This process allows the input device to be controlled by the microprocessor simply by keying its data into the channel.

**INTE A**
Controlled by setting/resetting the bit on PC4

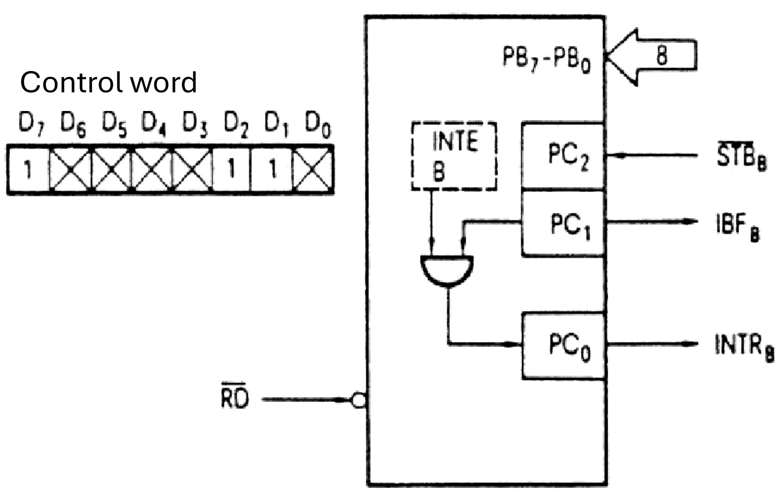
**INTE B**
Controlled by setting/resetting the bit on PC2

## Operating mode 1 — Inputs

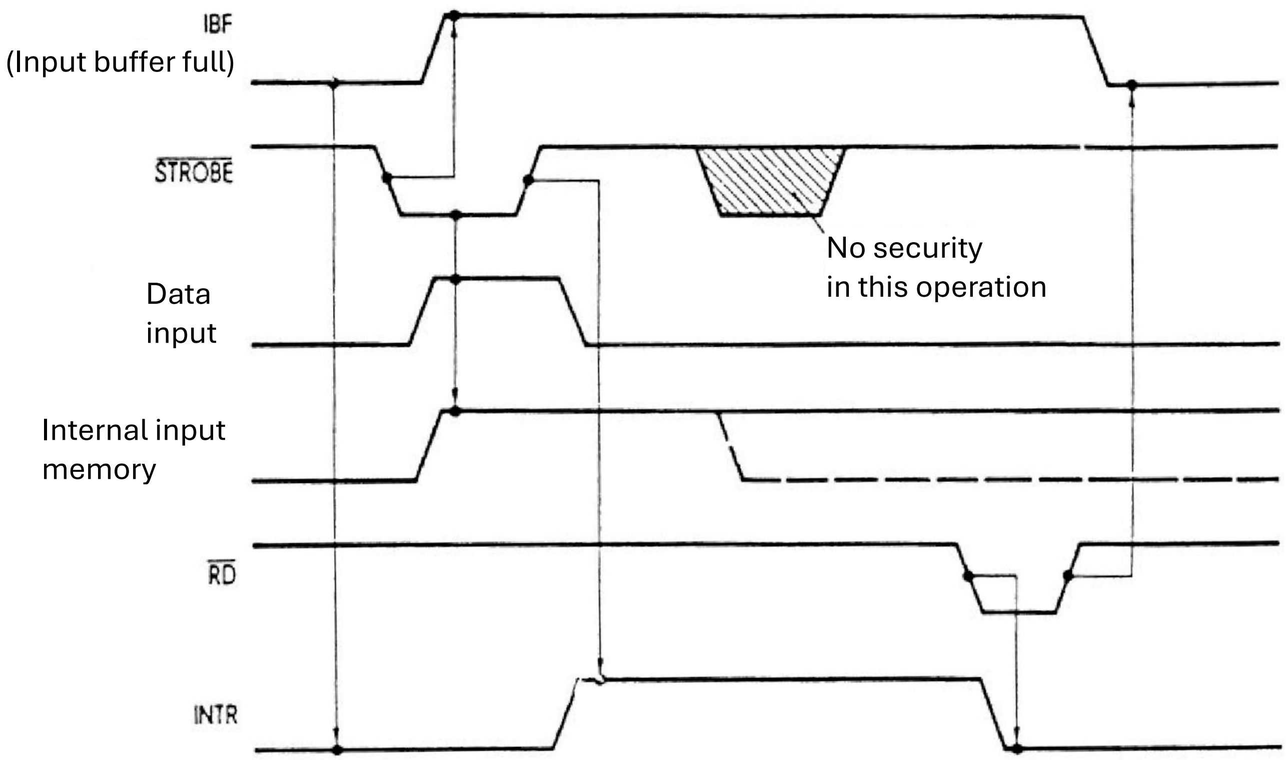
**Channel A**




**Channel B**



## Timing diagram for input (in principle)



## Definition of the output control signals

**/OBF (Output Buffer Flip-Flop Loaded)**
The /OBF output goes low when the microprocessor has written data to the selected channel. The /OBF flip-flop is set by the rising edge of the WR input and reset by the falling edge of the ACK signal.

**/ACK**
(Acknowledge Input) A LOW level at this input indicates to the SAB 8255 that data is being received from channel A or B. This signal is a response from the peripheral device confirming receipt of the data output by the microprocessor.

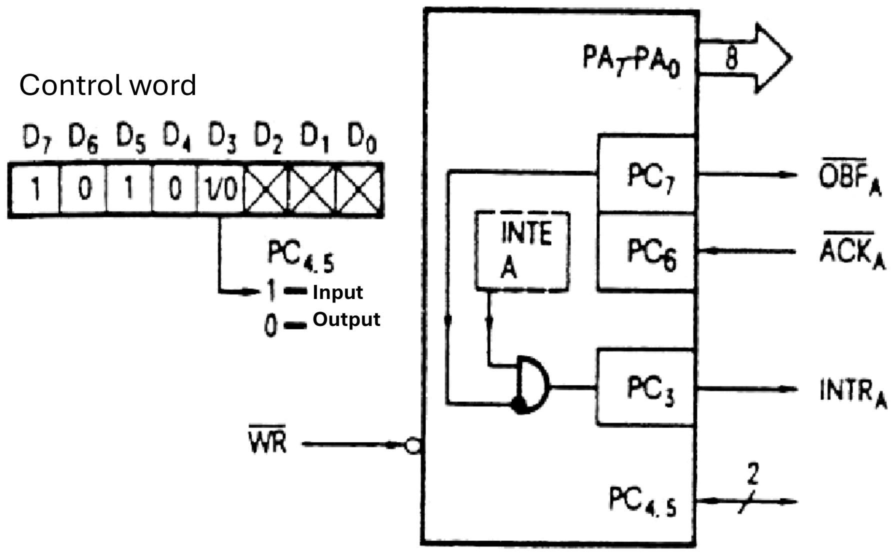
**INTR (Interrupt Request)**
A HIGH level at this output can be used to interrupt the microprocessor when the output device has accepted the data sent by the microprocessor. INTR is set by the rising edge of ACK when OBF is set to "one" and INTE is set to "one" at the same time. It is reset by the falling edge of WR.

**INTE A**
Is controlled by setting/resetting the bit on PC6

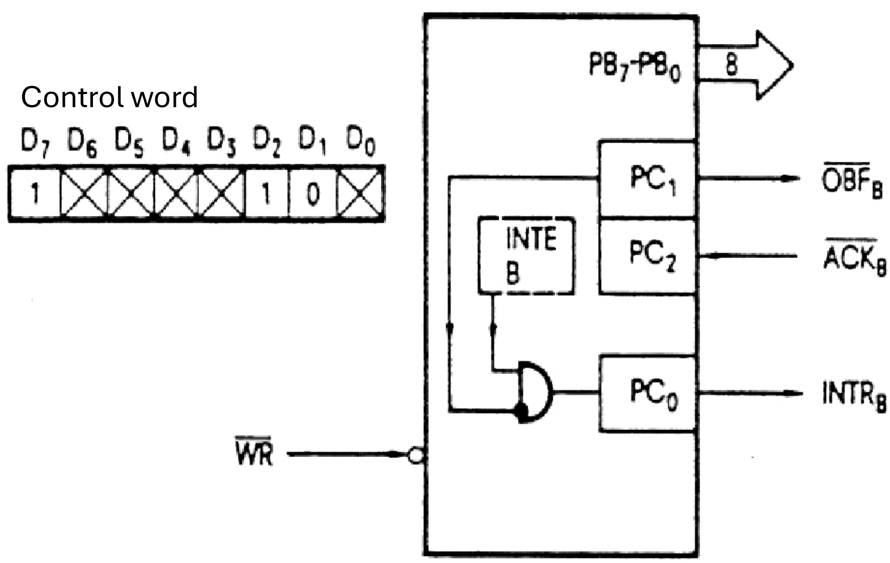
**INTEB**
Is controlled by setting/resetting the bit on PC2

## Operating mode 1 — Outputs

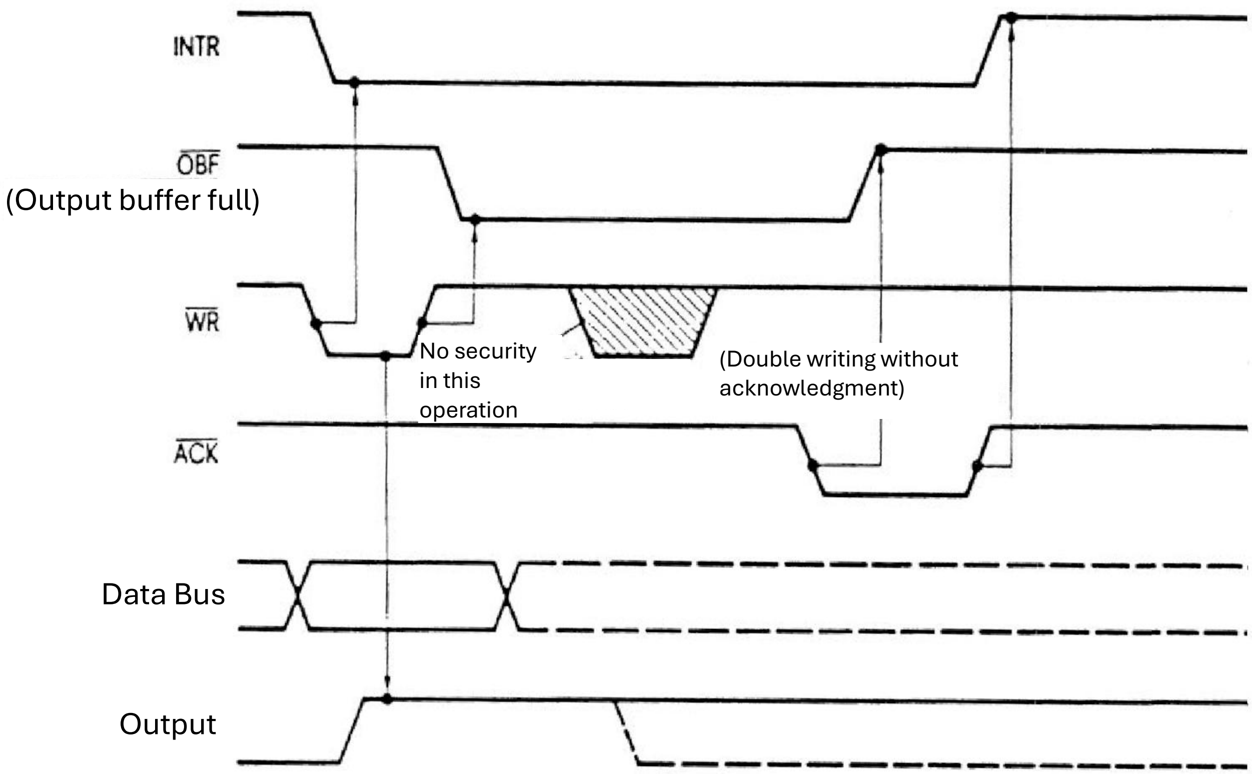
### Channel A




### Channel B

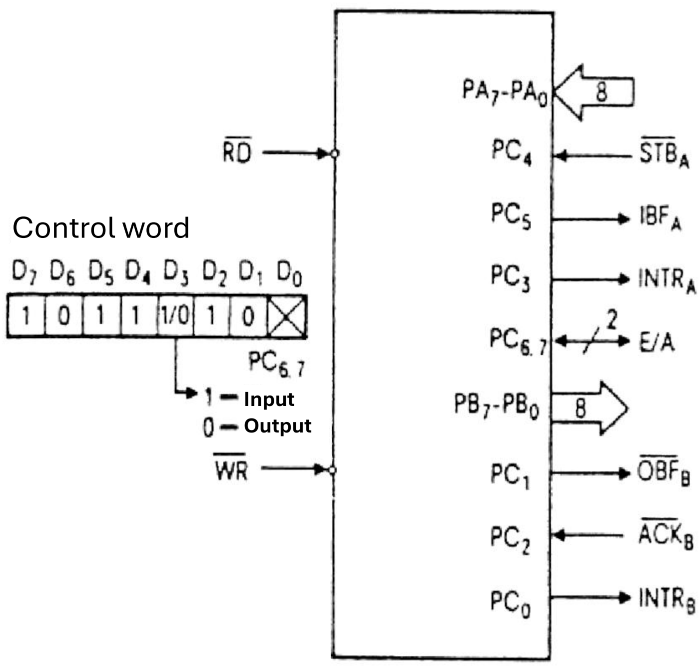


## Timing diagram for output (principle)

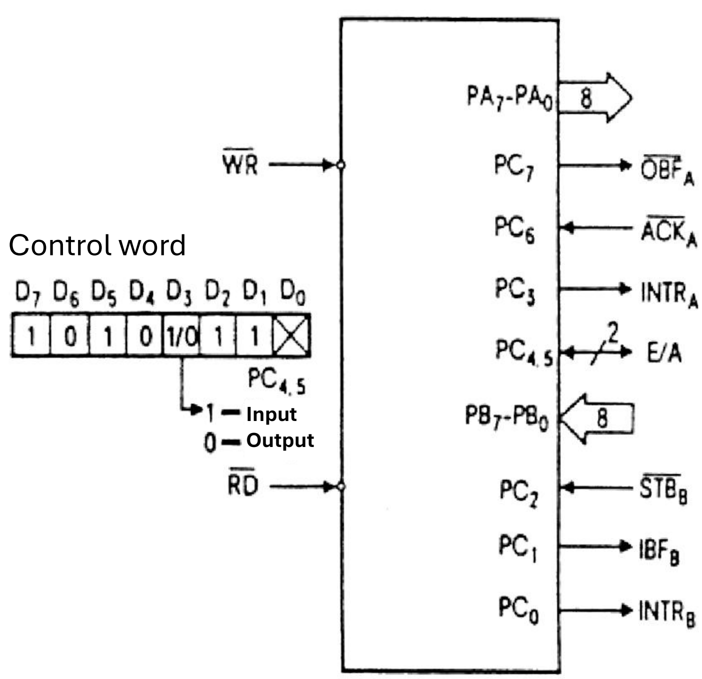


## Combinations in Mode 1

Channels A and B can be defined independently as inputs or outputs in Mode 1, enabling a variety of keyed I/O applications.




Channel A (keyed input)
Channel B (keyed output)




Channel A (keyed input)
Channel B (keyed output)

## Operating mode 2 (keyed two-way BUS input/output)

This functional arrangement enables data exchange with a peripheral device or circuit on a single 8-bit bus, over which data is sent and received (two-way bus input/output). Correct data flow on the bus is ensured by acknowledgement signals, similar to mode 1. Interrupt generation and lock/unlock functions are also available.

- Basic functional definitions of operating mode 2:
- Used only in group A
- One 8-bit two-way bus channel (channel A) and one 5-bit control channel (channel C)
- Inputs and outputs have buffers.
- The 5-bit control channel (channel C) is used for control and status purposes for the 8-bit two-way bus channel (channel A).

## Definition of the control signals for the two-way BUS input/output

INTR (Interrupt Request)
A HIGH level at this output can be used to interrupt the main program during input and output.

**Issue operation**

**/OBF (Output Buffer Loaded)**
The OBF output goes HIGH when the microprocessor has written data to channel A.

**CK (Acknowledge)**
A LOW level at this input enables the tri-state output buffer of channel A to send data. Initially, the output buffer is in the high-impedance state (3rd state).

**INTE1 (INTE flip-flop associated with /OBF)**
Controlled by setting/resetting the PC6 bit

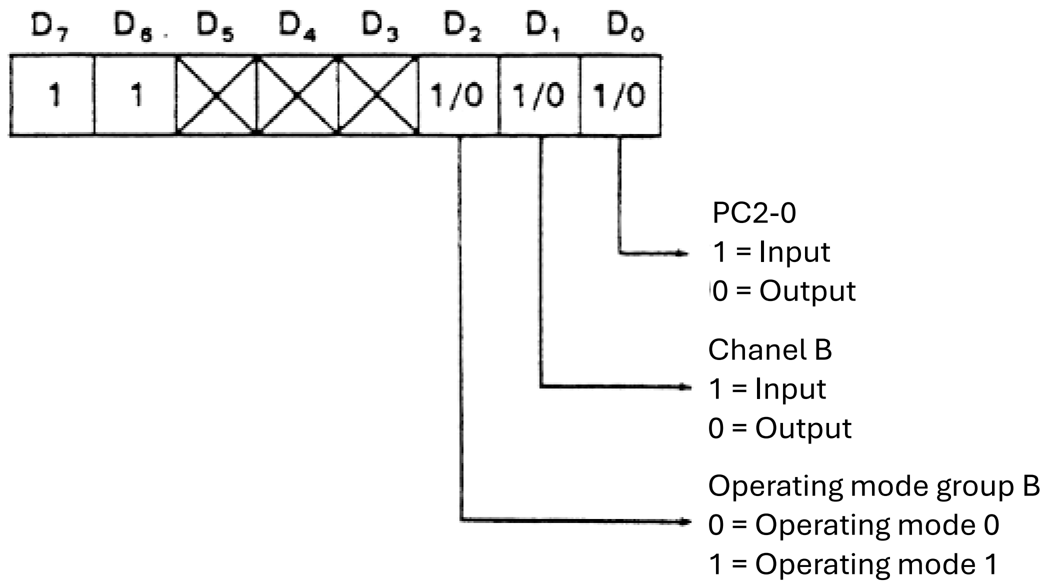
## Input mode

**STB (key input)**
A LOW level at this input loads data into the input buffer.

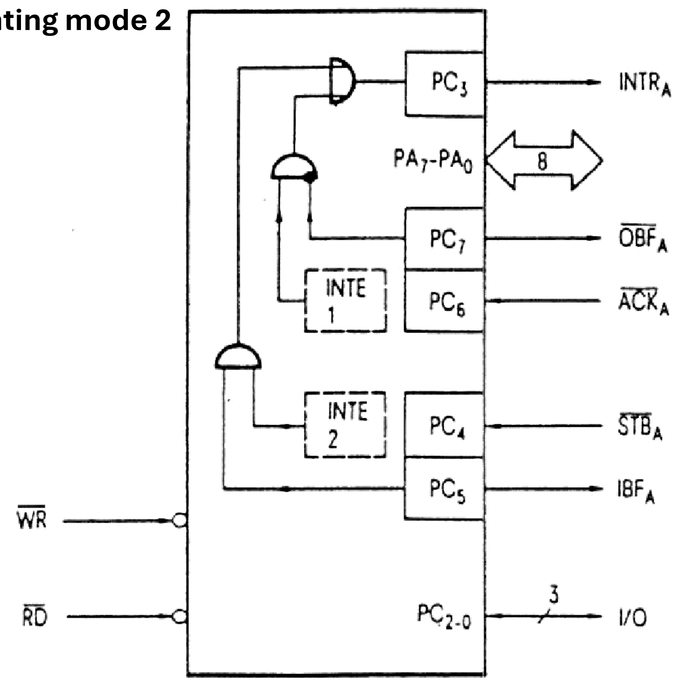
**IBF (Input Buffer Full Flip-Flop)**
A HIGH level at this output indicates that data has been loaded into the input buffer.

**INTE 2 (INTE flip-flop associated with IBF)**
Controlled by setting/resetting the PC4 bit.

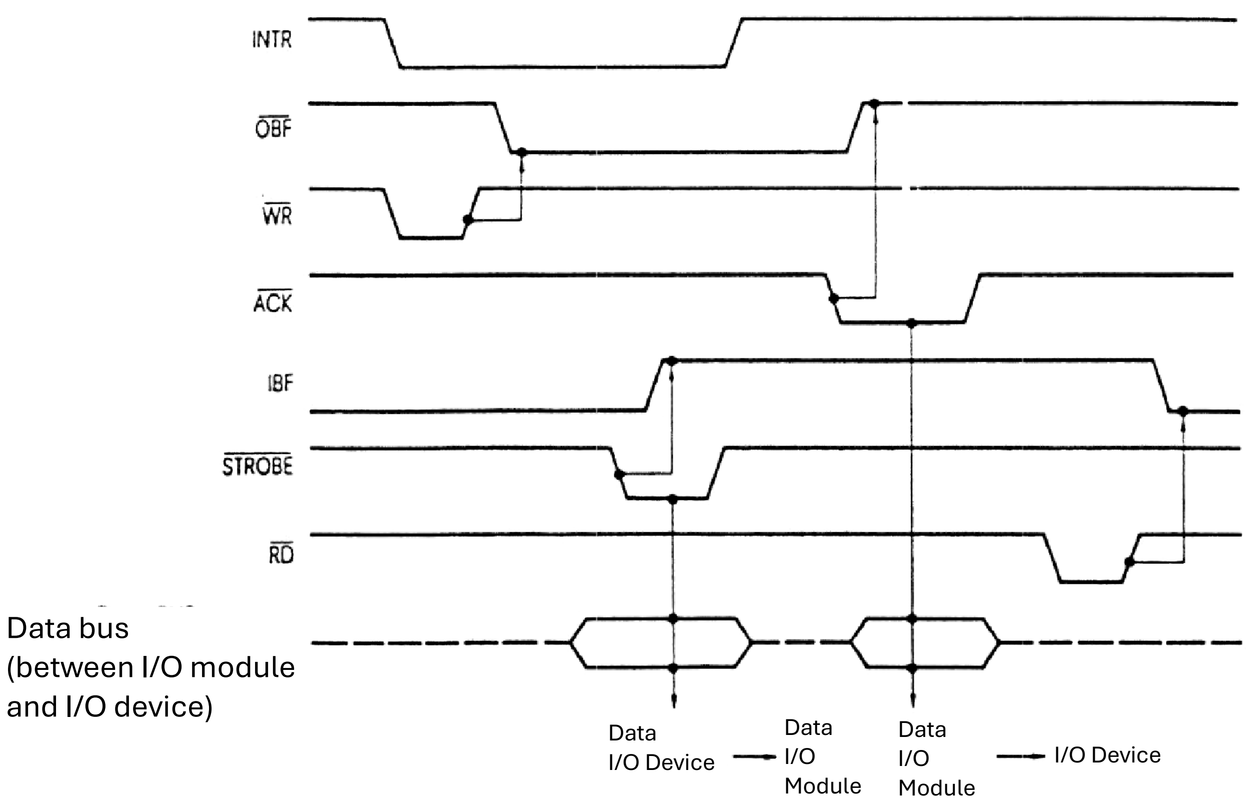
**Control word for operating mode 2**



**Operating mode 2**

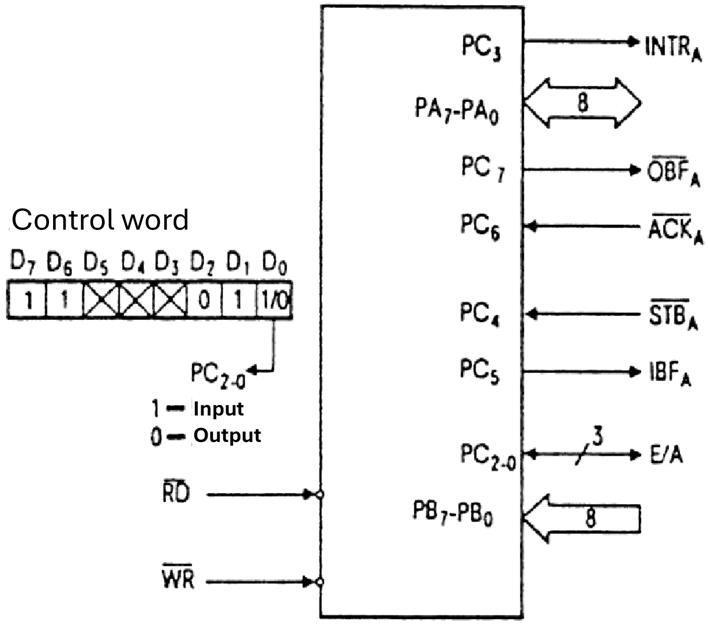



**Timing Diagram for Operating mode 2 (bi-directional)**

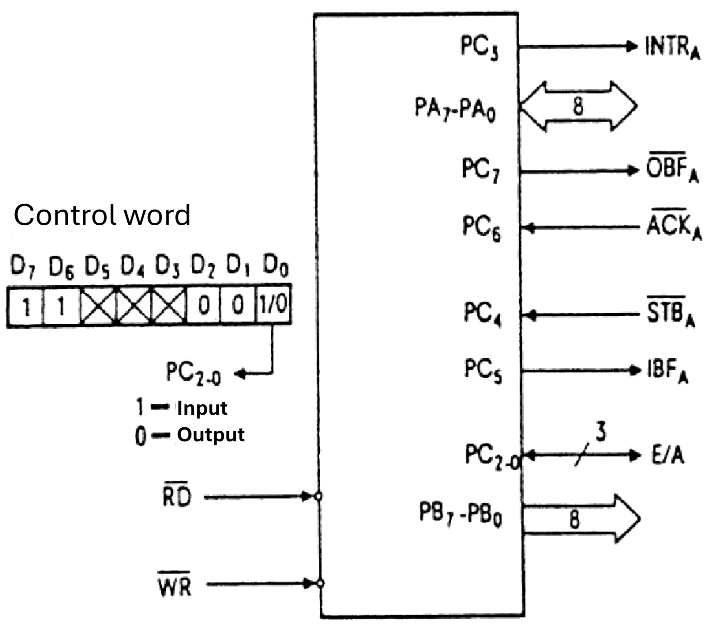


## Operating mode 2 — Combinations

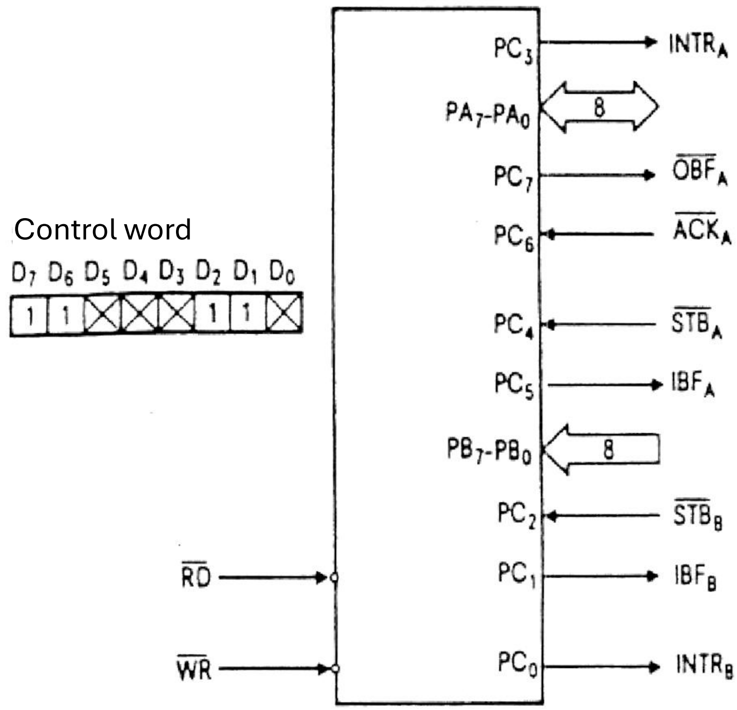
**Operating modes 2 and 0 (input)**



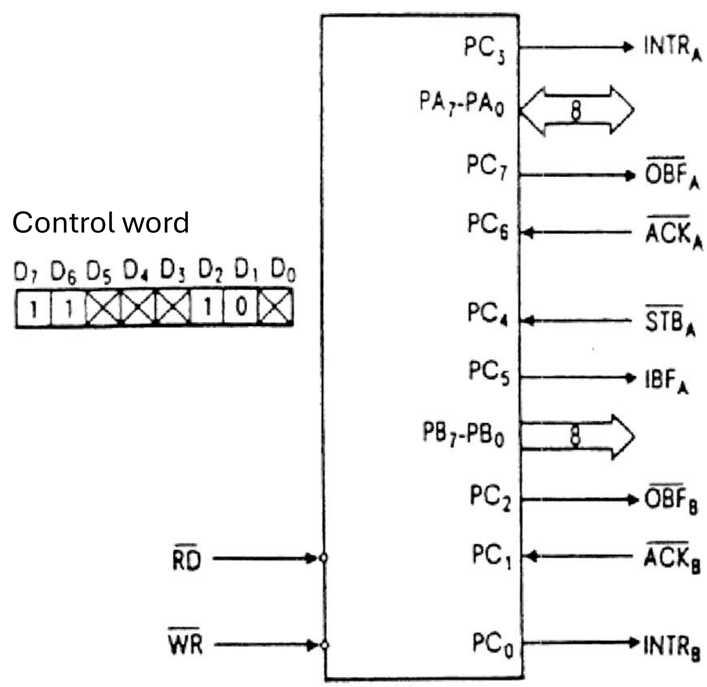

**Operating modes 2 and 0 (output)**



**Operating modes 2 and 1 (input)**




**Operating modes 2 and 1 (output)**



**Overview table of operating modes definition**

| | Operating mode 0 | | Operating mode 1 | | Operating mode 2 |
|---|---|---|---|---|---|
| | Input | Output | Input | Output | Group A only |
| PA0 | Input | Output | Input | Output | <-> |
| PA1 | Input | Output | Input | Output | <-> |
| PA2 | Input | Output | Input | Output | <-> |
| PA3 | Input | Output | Input | Output | <-> |
| PA4 | Input | Output | Input | Output | <-> |
| PA5 | Input | Output | Input | Output | <-> |
| PA6 | Input | Output | Input | Output | <-> |
| PA7 | Input | Output | Input | Output | <-> |
| PB0 | Input | Output | Input | Output | -- Only modes 0, 1 |
| PB1 | Input | Output | Input | Output | -- Only modes 0, 1 |
| PB2 | Input | Output | Input | Output | -- Only modes 0, 1 |
| PB3 | Input | Output | Input | Output | -- Only modes 0, 1 |
| PB4 | Input | Output | Input | Output | -- Only modes 0, 1 |
| PB5 | Input | Output | Input | Output | -- Only modes 0, 1 |
| PB6 | Input | Output | Input | Output | -- Only modes 0, 1 |
| PB7 | Input | Output | Input | Output | -- Only modes 0, 1 |
| PC0 | Input | Output | INTRB | INTRB | I/O |
| PC1 | Input | Output | IBFB | /OBFB | I/O |
| PC2 | Input | Output | /STBB | /ACKB | I/O |
| PC3 | Input | Output | INTRA | INTRA | INTRA |
| PC4 | Input | Output | /STBA | I/O | /STBA |
| PC5 | Input | Output | IBFA | I/O | IBFA |
| PC6 | Input | Output | I/O | /ACKA | /ACKA |
| PC7 | Input | Output | I/O | /OBFA | /OBFA |

**Notes for combinations of special operating modes**

In various combinations of operating modes, not all bits of channel C are used for control or status purposes. The remaining bits can be used as follows:

When programmed as input:
Access to all input lines is possible during a normal channel C load operation.

When programmed as an output:
Access to the higher-order bits of channel C (PC7-PC4) must be done individually using the bit set/reset function.

Access to the low-order bits of channel C (PC3-PC0) can be done with the bit set/reset function or by writing to channel C.

**Output power of channels B and C:**

Each set of eight output buffers, randomly selected from channels B and C, can deliver 1mA at 1.5V. This feature allows the SAB 8255 to directly drive Darlington drivers and high-voltage indicator units that require such currents.

**Reading the state of channel C**

Channel C transmits data to or from peripheral devices in mode 0. When the SAB 8255 is programmed for modes 1 or 2, the SAB 8255 generates or receives acknowledgement signals from the peripheral devices. By evaluating the contents of channel C, the programmer can test or verify the status of each peripheral device and modify the program flow accordingly.

Reading the status information from channel C is performed without a special command. A normal C-channel read operation performs this function.

Format of the status word
for operating modes 0 or 1

**Entrance arrangements**

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| J/0 | J/0 | $IBF_A$ | $INTE_A$ | $INTR_A$ | $INTE_B$ | $IBF_B$ | $INTR_B$ |
| **Group A** | | | | | **Group B** | | |

**Exit arrangements**

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| $\overline{OBF}_A$ | $INTE_A$ | 1/0 | 1/0 | $INTR_A$ | $INTE_B$ | $\overline{OBF}_B$ | $INTR_B$ |
| **Group A** | | | | | **Group B** | | |

**Status word format**
**for operating mode 2**

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|---|---|---|---|---|---|---|---|
| $\overline{OBF}_A$ | $INTE_1$ | $IBF_A$ | $INTE_2$ | $INTR_A$ | | | |
| **Group A** | | | | | **Group B** | | |

**(Defined by selecting operating modes 0 or 1)**

## Static characteristics

Tu = 0 to +70°C; Vcc = +5V +-5%; GND = 0

| Symbol | Designation | Limit values | | | unit | Test conditions |
|---|---|---|---|---|---|---|
| | | Min | Typ. | Max | V | |
| Vil | L input voltage | - | - | 0.8 | | - |
| Vih | H input voltage | 2 | - | - | | |
| Vol | L-output voltage | - | - | 0.4 | | Iol=1.6ma |
| Voh | LH-output voltage | 2.4 | - | - | | Ioh=-50us (-100ua for data bus channel |
| Ioh | Darlington driver current | - | 2 | - | ma | V0h=1.5V, Rext-390ohm |

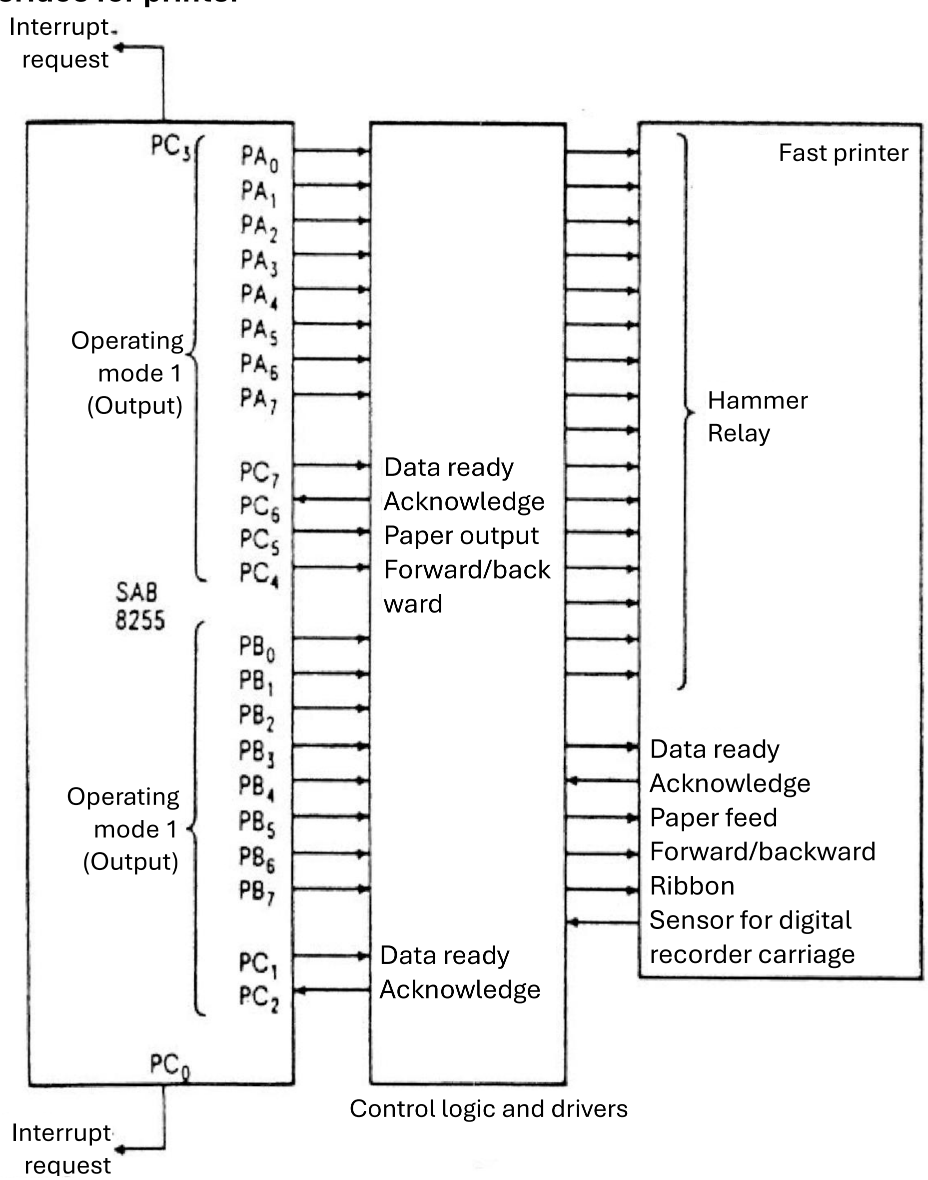
## Applications of the SAB 8255

The SAB 8255 is a very powerful module for connecting peripheral devices to the SAB 8080. It makes optimal use of the existing connections and is flexible enough to connect many I/O devices without additional circuitry.

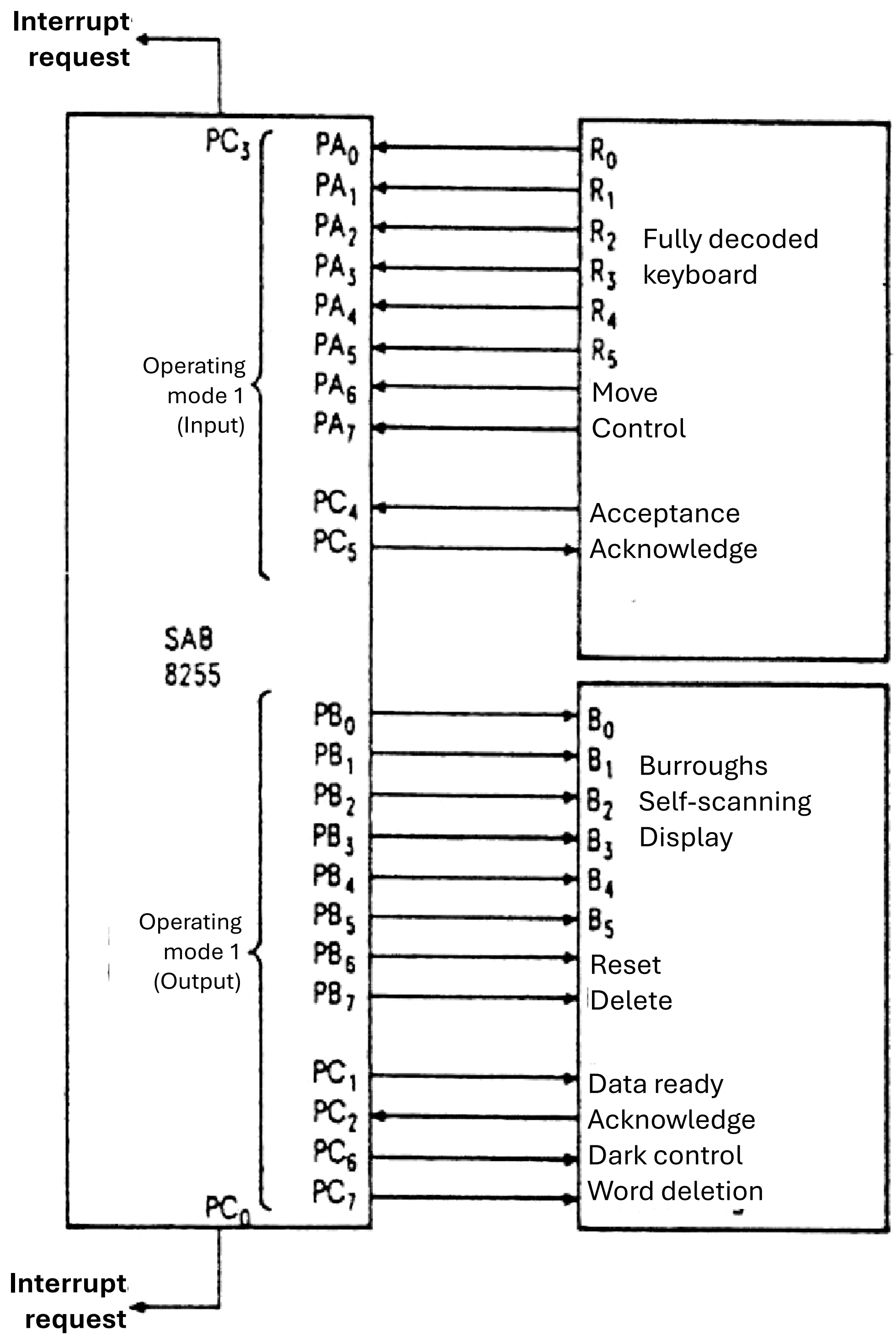
Each peripheral device is normally assigned a "utility program" in the microcomputer system. This program manages the software interface between the device and the microprocessor. The functional definition of the SAB 8255 is programmed by the I/O utility program and represents an extension of the system software.

Some typical applications of the SAB 8255 are listed here.

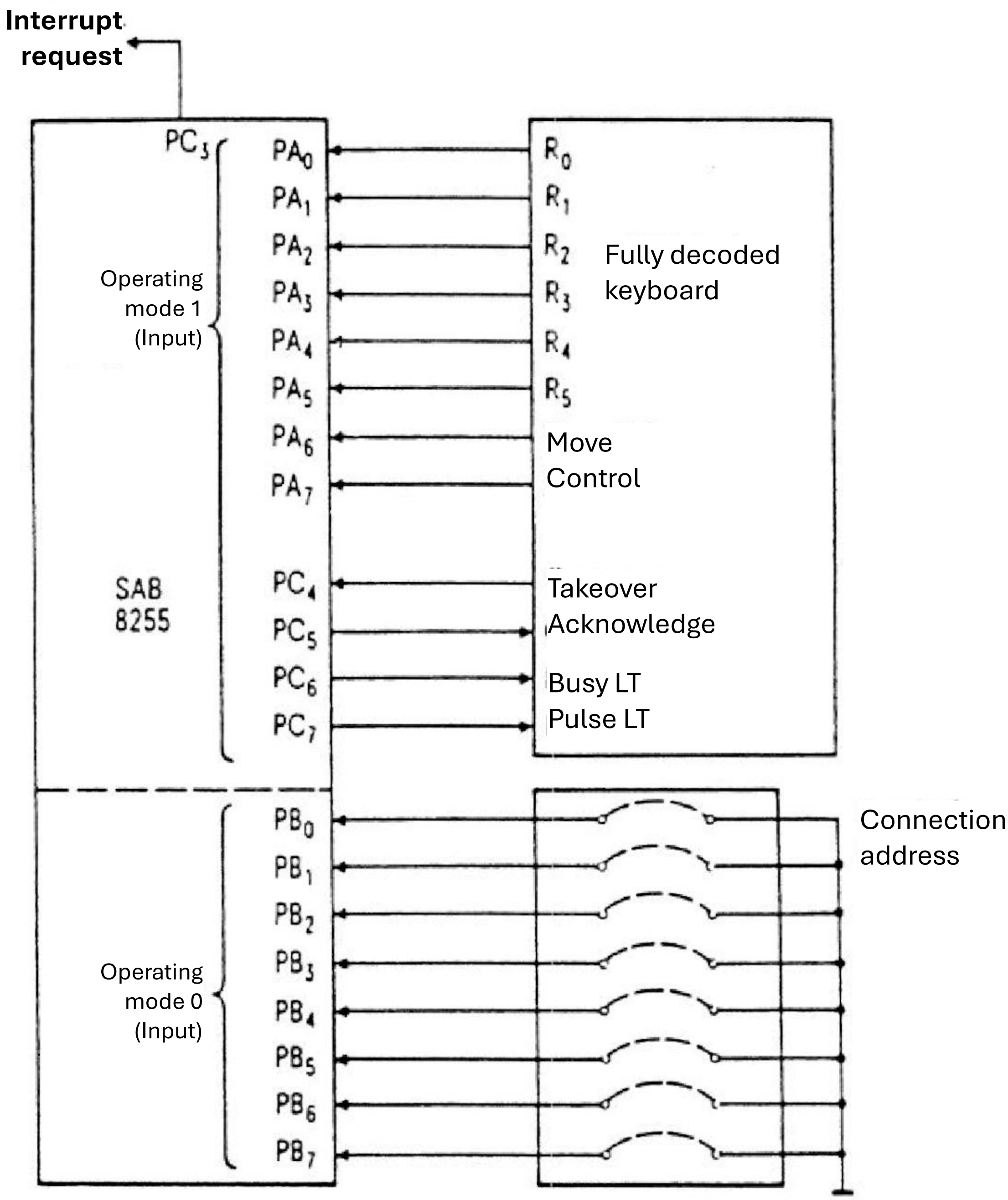
**Interface for printer**



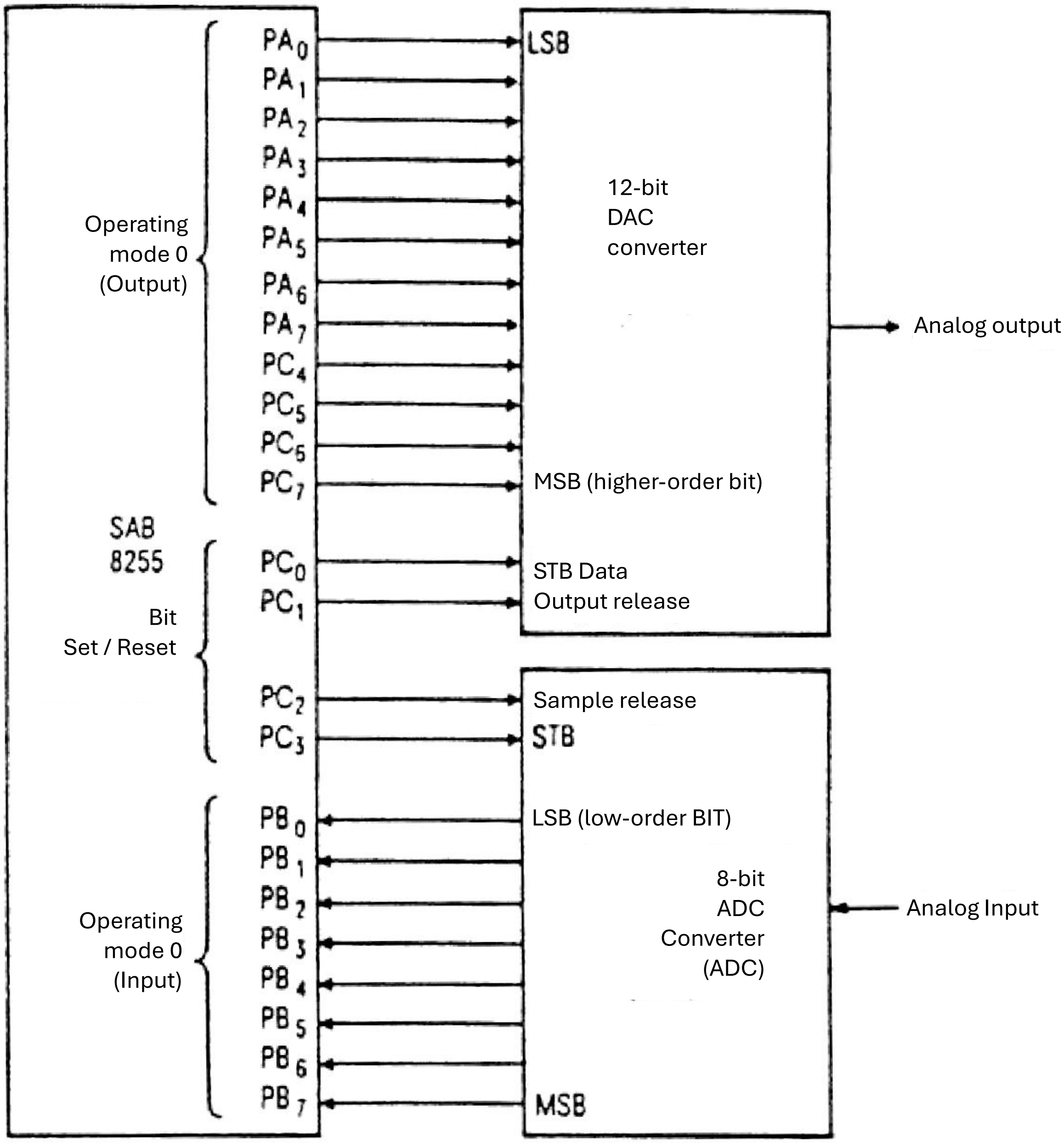
### Interface to keyboard and display device



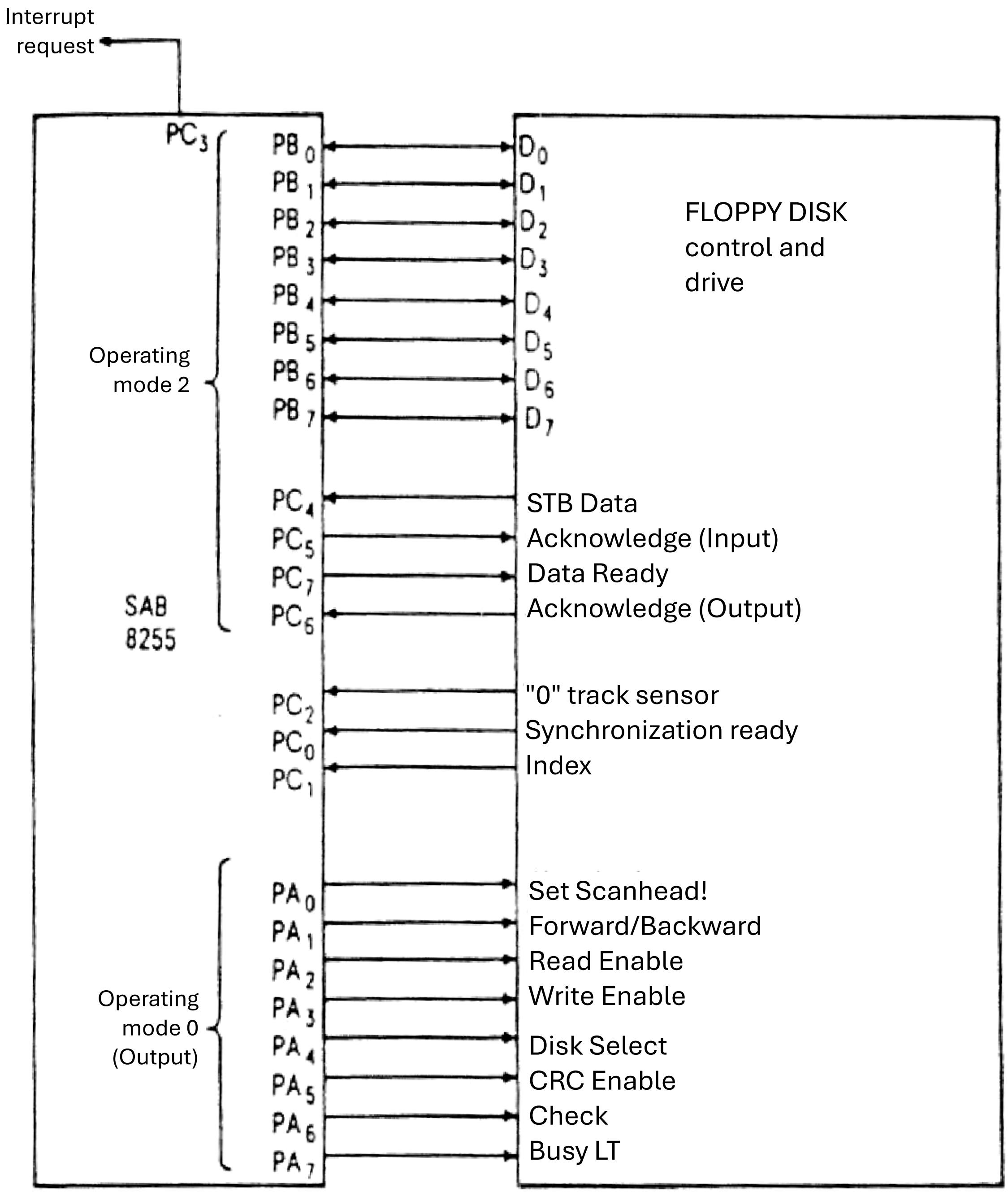
## Interface to keyboard and connection addresses



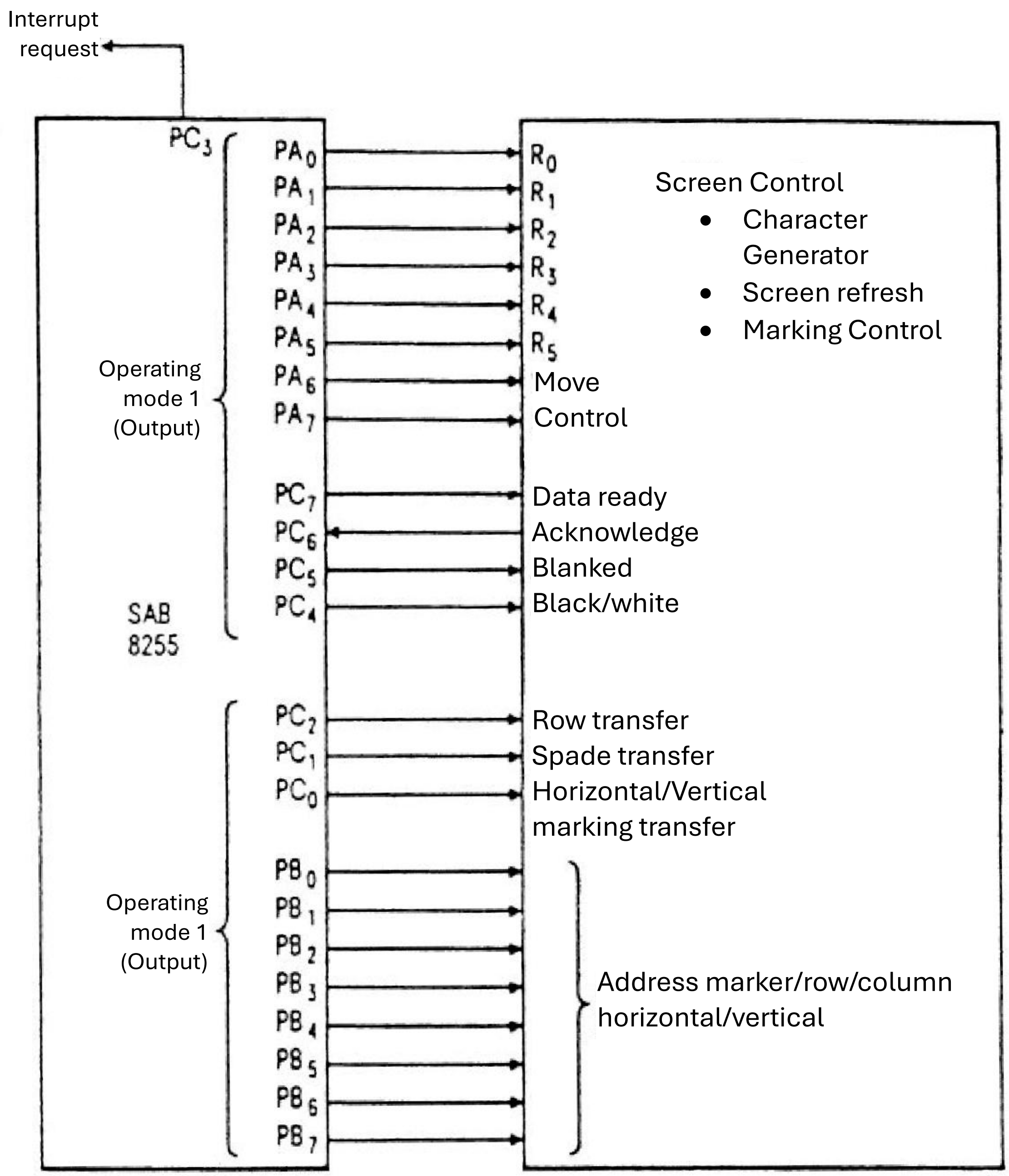
## Interface to digital/analog and analog/digital converters



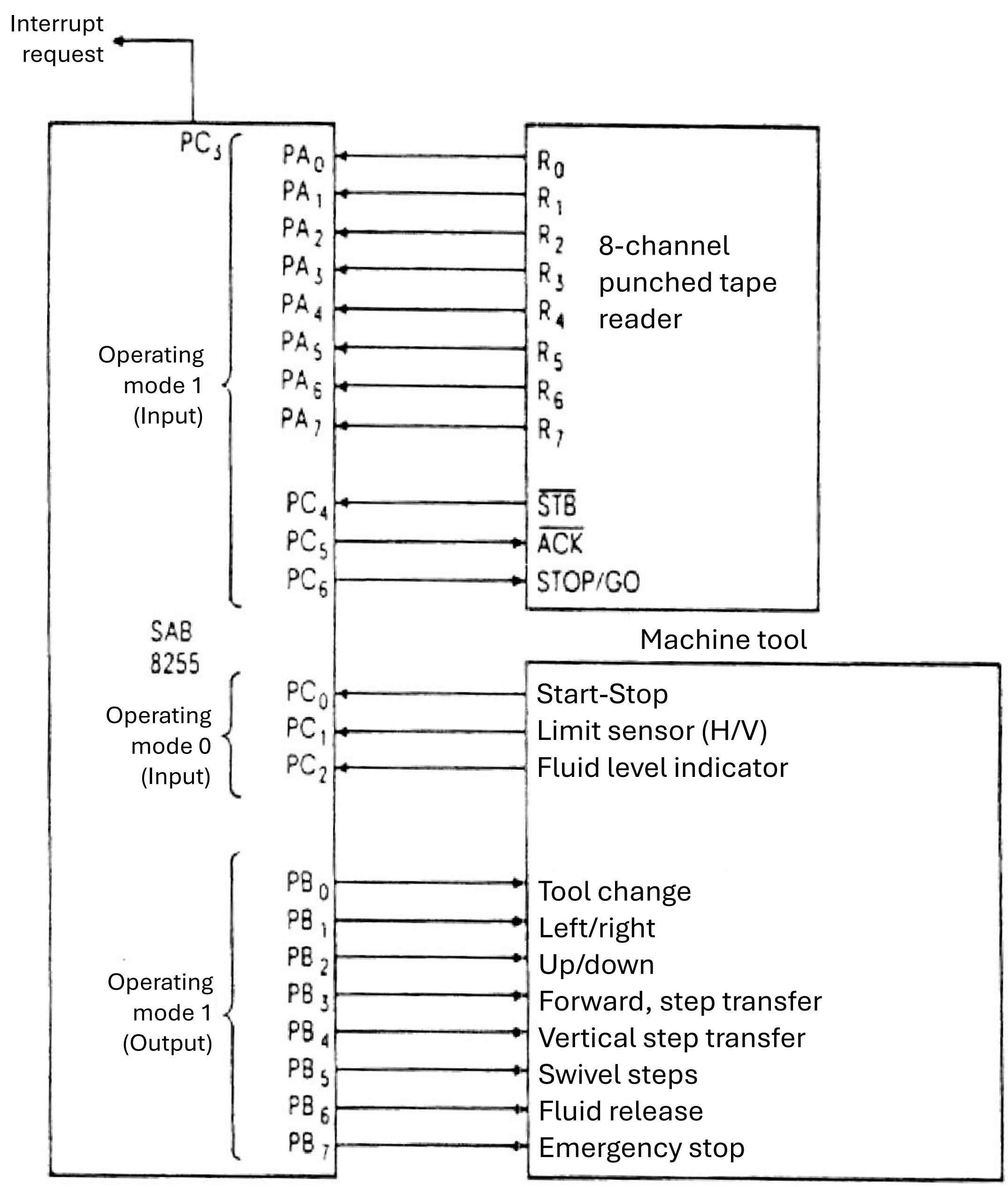
## Interface to floppy disk control and drive



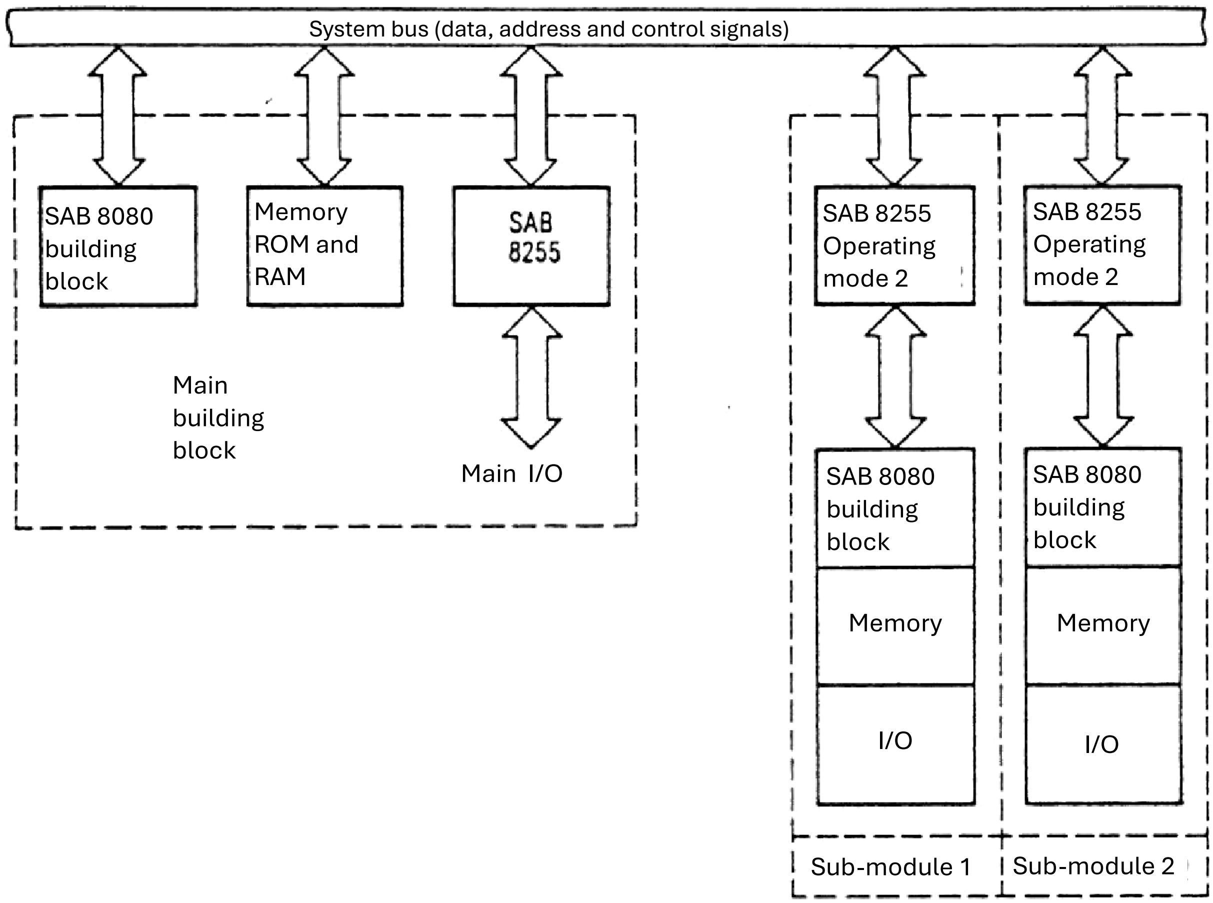
### Interface to screen control



## Interface to a machine tool



## Interface to a multi-processor ('decentralized intelligence')



# Options

**Table of contents**

# MC 12.1
# 11-BIT / MEMORY EXPANSION

**11-bit measurement value acquisition**

**`DINSCAN`**
**`DINSCAN N`**
Abbreviation: **`DIN. DIN.N`**

A transient measurement can be triggered analogously to the **`INSCAN`** command. All parameters are set as with the standard **`INSCAN`** command. The data is only sampled with 11-bit accuracy if double-precision measurement buffers are configured with the **`DBUFINIT`** command.

If an 11-bit measurement buffer is configured (using the **`DBUFINIT`** command), the following applies:

The measured values are acquired with 11-bit accuracy.

By appending **`N`** to the command, the measurement starts without waiting for the trigger point.

If you abort the measurement using the <BREAK> button:
The acquired data remains in the MC-12(A)'s memory.

The minimum sampling time is longer than with **`INSCAN`**:

| | | | | | |
|---|---|---|---|---|---|
| Number of channels: | 1 | 2 | 3 | 4 | 5 |
| Sampling time: | 0.8 | 1.4 | 2.0 | 2.6 | 3.2 ms |

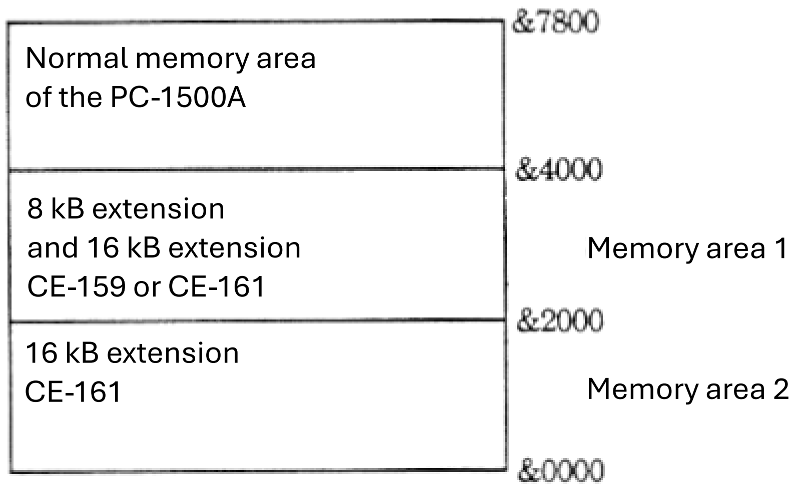
As soon as a trigger event is detected, the special character *ht* is displayed on the LCD. This character is deleted after the measurement is completed or after a <BREAK>.

History cannot be recorded.

**Memory expansion:**

Using the following three commands, it is possible to buffer all MC-12(A) data in an 8 or 16-KB RAM/ROM module. However, due to the different memory allocations, this is only possible with the PC-1500A.

Memory structure when using these commands:




**`MSAVE`** <number>
Abbreviation: **`MS.`** <number>

The entire contents of the MC-12(A) are stored in the memory with the specified number:

Number=1: Memory from address &2100 to &3FFF
Number=2: Memory from address &0100 to &1FFF (not possible with 8-KB module)

The values in the MC-12 memory remain unchanged. The reserve allocations also remain unchanged. Saving only occurs if no program is located in this area. The lack of changeable memory (RAM) is indicated by a corresponding error message (**`ERROR 137`**).

**MLOAD** <number>
Abbreviation: **ML.** <number>

In reverse of the **MSAVE** command, the MC-12 memory is reloaded with the saved data from the specified memory area. The data can also be loaded from non-modifiable memory (ROM). This allows the data to be protected by switching the write-protect switch on the RAM/ROM module.

Caution: If no data has been previously stored in this memory, the MC-12(A) will be incorrectly initialized. Either valid data must be loaded or the MC-12(A) memory must be initialized using the INIT command.

**MSWAP** <number>
Abbreviation: **MSW.** <number>

This command swaps the MC-12(A) memory with the specified memory. This allows a 16 KB RAM/ROM module to process three times the MC-12(A) memory. After saving the data using **MSAVE**, the individual data records can be swapped with the MC-12(A) data memory using the **MSWAP** command.

**Error messages:**
Here is a summary of all possible errors:

**ERROR 1**: Incorrect command input or missing numeric value
**ERROR 19**: Numeric value not within the permitted range
**ERROR 101**: MC-12 is not turned on (enter MCON command)
**ERROR 110**: Trigger level outside the measurement range
**ERROR 111**: Specified SCANTIME is too low
**ERROR 125**: With an 8-bit buffer (BUFINIT command), no measurement is possible without a trigger.
**ERROR 135**: Voltage in the MC-12(A) is no longer sufficient.
**ERROR 136**: A program is located in the specified memory area.
**ERROR 137**: Saving and data exchange are only possible with RAM.

# MC 12.M
# 24 - CHANNEL MULTIPLEXER

**Operation**

Your device has a 6th input channel, which is directly connected to the built-in 24-channel multiplexer. Using the MUX (channel) command, one of the 24 multiplexer channels can be switched to the 6th input channel. The 6th input channel is treated by the MC-12(A) in the same way as the 5 standard inputs of the MC-12(A). The 6th channel can therefore be selected both in multimeter mode (key 6) and in transient mode using the SELECT command. The multiplexer channel selected with the MUX (1...24) command is displayed or recorded. The multiplexer channel cannot be changed during an INSCAN command.

**Technical specifications**

Measuring range: +5V...4+5mV
Input impedance: 1MΩ MC-12 off 1kΩ

**Command syntax**

**MUX** (channel number 1..24)

Pinout of the 25-pin input connector

The following pinout applies to the second 25-pin connector on the rear of the MC-12(A):

Pins 1-24 correspond to channels 1-24
Pin 25 common ground

**Program Example**

```
10 REM SAMPLING 24 CHANNELS VIA MULTIPLEXER
20 TEXT:CSIZE1:FOR 1=1T024
30 MUX I:A=CHA6
40 USING "##,###":LPRINT"Channel";I;" ";A;" V"
50 NEXT I
```

# MC 12.Z
# COUNTER

Starting and querying the built-in counters. The command always waits for the next second edge of the built-in clock before executing the command. This allows for high precision in pulse counting.

Command: **ZRUN**
Syntax: **ZRUN**

Reset both counters to zero and start counting incoming pulses.

Command: **ZREAD**

Syntax: **ZREAD** Variable
**ZREAD** Variable, Variable

Read one or both counters. Both counters are simultaneously reset and restarted. This allows, for example, the counter reading to be queried continuously every 10 seconds.

Command: **ZWAIT**
Syntax: **ZWAIT** number

Waits for a specified number of second edges. The time range is 0 to 65535 seconds.

Command: **ZTOR**

Syntax: **ZTOR** Count, Variable
**ZTOR** Count, Variable, Variable

Combines all three commands. The counters are started on the next second edge. After waiting for the specified time (in seconds), the counters are read. This allows for a single pulse capture over a selected period of time.

# MC 12.T1
# Function Generator

Output of signals on the analog output X1 (and X2) using BASIC commands:

**`BUFOUT:`**
Output of the entire buffer with or without repetition.
Output channels: X1 and X2
Output frequency: 0.763 to 5726 sample points per second

**`SIGOUT`**:
Output of a range with repetition.
Output channel: X1
Output frequency: 0.003 to 13830 Hertz for the entire range

**Applications:**

Playback of a recorded signal at different frequencies (e.g., voice output for several seconds)

Output of an artificial signal

Parallel output of two buffers for X-Y recorders

| | | |
|---|---|---|
| Error: | **`ERROR 1`** | - Incorrect command input |
| | **`ERROR 39`** | - Invalid numeric value |
| | **`ERROR 101`** | - MC12(A) is not turned on (enter **`MCON`**) |

**1. BUFOUT: Output of an entire buffer**

Syntax: **BUFOUT** f, b1 ('b2) (;R)

f: Frequency (number of measured values per second)
Maximum frequency: 5726 Hertz
Minimum frequency: 0.763 Hertz

bl: Number of the buffer that is output to analog output X1.

b2: Number of the buffer output at analog output X2. If omitted, only analog output X1 is used.

**;R**: If a "**;R**" is appended to the command, the output of the entire buffer is repeated until the <BREAK> key is pressed.

The entire buffer is output sequentially at the specified frequency to the analog outputs X1 (and X2). Signal output can be interrupted prematurely by briefly pressing the <BREAK> button. The program then continues normally. Holding the <BREAK> button down for too long will abort the program. However, the **CONT** command can be used to restart the program from the interruption point.

Cancel the command: Press the <BREAK> key.

Voltage at outputs X1 and X2:

Unipolar operation: 0 volts to 4.88 volts
Bipolar operation: - 4.96 volts to 4.88 volts

**2. SIGOUT: Output of an area**

Syntax: SIGOUT f,b,a,e

f: Frequency at which the entire range is output. Maximum frequency: 13830 Hertz (with 2 individual values) Minimum frequency: 0.003 Hertz (with 256 individual values)

b: Number of the buffer in which the area is located.

a: Number of the first point.

e: Number of the last point (a<e).

The specified range is standardized to 2 to 256 individual values (depending on the frequency) using linear interpolation. These individual values are then output at the specified repetition frequency at analog output X1. This makes it possible to output a signal once generated at different frequencies.

Cancel the command: Press the <BREAK> key.

Voltage at outputs X1 and X2:
Unipolar operation: 0 volts to 4.88 volts
Bipolar operation: -4.96 volts to 4.88 volts

Number of individual values:

The number of individual values generated by interpolation is predetermined by the specified frequency. This is due to the high cutoff frequency and the desired small error. The following formula is used to determine the number of individual values:

Number of individual values: $\frac{T}{47*F}$ T: Clock frequency (1.3 MHz) F: Specified frequency (Hz)

At a frequency below 108 Hertz, 256 individual values are always output. The formula given above also results in a maximum frequency of 13830 Hertz with 2 individual values.

Selection of the range for given and generated signals:

The **SIGOUT** command is actually intended to output a recorded signal at a changed frequency. To do this, the position numbers of identical points in adjacent periods of the signal are determined. Although the voltage value of the last point is required for interpolation, this value itself is not output, as it is equal to the first measured value. This should be taken into account when outputting generated signals:

- Select a buffer length equal to or greater than 2 blocks.
- Use 256 individual values for the entire signal.
- Set the 257th individual value to the value of the first point.
- If possible, start with the maximum or the first value after an edge.
  (e.g., cosine oscillation instead of sine oscillation)

The most important point is the use of a total of 257 predefined values, since otherwise the desired real-wave voltage would become a triangular or sawtooth wave.

Example 1: Output of a square wave with maximum frequency:

```
10 INIT B : BUFINIT 5.5 : BUFOPEN 1,4.88
20 FOR I=1 TO 128
30 BUFWRITE 1,1,4.88 : BUFWRITE 1,1+128,-4.92
40 NEXT I
50 BUFWRITE 1,257,4.88
60 SIGOUT 13830,1,1,257
```

Example 2: Output of a cosine waveform with a specified frequency:

```
10 INIT B : BUFINIT 5.5 : BUFOPEN 1.4.88
20 FOR I=0 TO 255
30 BUFWRITE 1.1+1.4.88 * COS( 1* 360/ 256 )
40 NEXT I
50 BUFWRITE 1.257,4.88 * COS(0)
60 INPUT "Frequency :';F : IF F<0.003 OR F>13830 THEN 60
70 WAIT 0 : PRINT "End with BREAK key"
80 SIGOUT 13830,1,1,257
90 GOTO 60
```

Example 3: Calculating the number of the start and end points:

To do this, the times at both points are recorded in SCREEN mode. The position number can then be determined using the following formula.

$$\text{Position number} = \text{PREHIST} * 256 + \frac{\text{measured time}}{\text{SCANTIME}}$$

**3. Information on frequency accuracy**

The frequency specified in the commands cannot normally be achieved exactly. This is due to program structure and computer inaccuracies.

**3.1 Quartz crystal frequency of the PC-1500(A)**

The PC-1500(A) computer operates at a frequency of 1300,000 Hz. This value is not the same for all devices, but can deviate by up to 0.5% from the target. Since this deviation generally remains constant, this inaccuracy can be corrected by applying a frequency correction factor.

$$\text{Correction factor} = \frac{\text{entered frequency}}{\text{Output frequency}}$$

This factor must be taken into account for frequencies below 50 Hertz, as the program-dependent error then drops below 0.5 percent.

## 3.2 Frequency inaccuracy due to the program structure

**SIGOUT**

In the frequency range from 13830 Hz to 108 Hz, the maximum error remains constant, as the number of points increases up to 256. From 108 Hz onwards, this error depends only on the desired frequency.

13830 Hertz to 50 Hertz: < 0.015 (corresponds to 1.5%)

50Hz to 0.003 Hz $\frac{\Delta F}{F} < \frac{256*F}{T+256*F}$

T: Clock frequency 1.3MH
F: Desired frequency (Hz)
ΔF: Frequency deviation (Hz)

| | | | | | |
|---|---|---|---|---|---|
| Examples: Desired frequency: | 50 | 10 | 1 | < | 0.1 Hertz |
| Maximum error: | 1 | 0.2 | 0.02 | < | 0.002 percent |

**BUFOUT**

Since the output is delayed by a waiting loop, which can be extended in steps of 26 clock cycles, the following calculation formula for the accuracy results:

5726Hz to 0.763 Hz $\frac{\Delta F}{F} < \frac{26*F}{T+26*F}$

T: Clock frequency 1.3MH
F: Desired frequency (Hz)
ΔF: Frequency deviation (Hz)

# MC 12.55
# PLOTTER PROGRAM MODULE

This module provides BASIC commands that facilitate graphical output of data on the CE-150 plotter:

- Drawing a coordinate grid with one command

- Drawing scales in the X and Y directions with:
  - freely selectable divisions
  - labels that can be placed independently of the divisions
  - optionally linear or logarithmic labels
  - compliance with the **`USING`** format during output

- Output of data from the MC-12 (A) and MC-10 (A)
  - Voltage range selectable independently of the **`BUFRANGE`** (zoom)
  - Interpolation of measured values for expanded output
  - Automatic processing of 8- or 11-bit measured values

**General description**

All commands are only applicable in **`GRAPH`** mode (otherwise **`ERROR 73`** is output).

All new plotter commands use a rotated coordinate system. The coordinate system is rotated so that the X-axis is pointing downward and the Y-axis is pointing right. When outputting measured values, the voltage axis (Y-axis) is drawn horizontally and the time axis (X-axis) vertically, similar to the BASIC command **`PLOT`**.

The output uses the most recently defined origin (zero point) of the normal coordinate system as the origin of the rotated coordinate system. Except for the **`MOVE`** command, all representations are output in the positive X direction (downward) starting from the zero point of the X axis.

All plotter outputs are only possible within a range of -1999 to +1999 units on both axes. If this range is exceeded by a command, the command is usually terminated by

All plotter outputs are only possible within a range of -1999 to +1999 units on both axes. If this range is exceeded by a command, **`ERROR 70`** is usually displayed before the command is executed. Please note that the offset and the Y-axis are additive.

All plotter commands listed below use the rotated coordinate system.

Only those parts of the graphic to be output that lie within the displayable range are plotted. No error message is generated as long as the pen does not move outside the range of -2048 to +2047 steps in the X and Y directions.

After switching from **TEXT** mode to **GRAPH** mode, the current pen position is the origin of the coordinate system. A configurable offset shifts the origin to the right (i.e., in the positive Y direction) for positive values.

It is recommended that you try out the given examples with different parameters to familiarize yourself with the diverse possibilities of these commands. The **GATE**, **XSCALE**, **YSCALE**, and **DRAW** commands are coordinated in such a way that the parameters o, x, and y can usually remain the same for graphic output.

### Command description

**MOVE** x, y
Abbreviation: **MOV.**

x : Units in the X direction (-1999 < x < 1999)
y : Units in the Y direction (-1999 < y < 1999)

Moves the pen to the specified position in the rotated coordinate system. No offset can be specified here. The **SORGN** command defines the current pen position as the new origin of the coordinate system.

**GATE** o,x,y,dx,dy,x1,x2
Abbreviation: **GA.**

o: y-coordinate of the lower left corner of the rectangle (-1999 ≤ o ≤ 1999)
x: units in the x-direction (-1999 ≤ x ≤ 1999)
y: units in the y-direction (0 ≤ y, y+o ≤ 1)

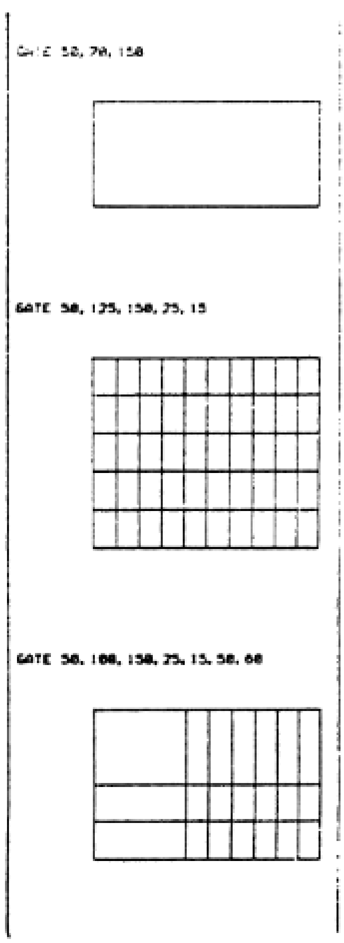
x:	Units in the X direction (-1999 ≤  x ≤  1999)
y:	Units in the y-direction (0 ≤  y, y+o ≤ 1)
x:	Distance between two lines perpendicular to the x-axis (0 ≤  dx ≤  65535)
dy:	 distance between two lines perpendicular to the y-axis (0 ≤  dy ≤  65535)
x1:	Distance of the 1st X-line from the Y-axis (0 ≤  xl ≤  65535)
y1:	Distance of the 1st Y-line from the X-axis (0 ≤  yl ≤  65535)

Drawing a coordinate grid with specified parameters. Unnecessary parameters can be omitted. All values are specified in plotter units (0.2 mm).

After switching to **GRAPH** mode, the pen is positioned on the far left. The last writable point is 215 steps to the right (positive Y-axis). Therefore, if an offset o greater than 215 is specified, the grid lies completely outside the paper strip.

All special cases for dx and x1 given below are described only for the X-axis, but are analogously valid for the Y-axis (dy and y1).

Example:		Syntax and description:




**GATE** o,x,y

Draw a rectangle with a specified length x and width y. A positive offset o moves the rectangle to the right.

**GATE** o,x,y,dx,dy

Draw a grid with line spacing dx in the rectangle. If dx is zero, no additional line is drawn perpendicular to the x-axis. With a step size of 1, the lines are drawn directly adjacent to each other, thus filling the entire area of the rectangle.

**GATE** o,x,y,dx,dy,xl, yl

When drawing the grid, the first line perpendicular to the x-axis starts at position x1. If x1 is greater than x, no further lines are drawn.

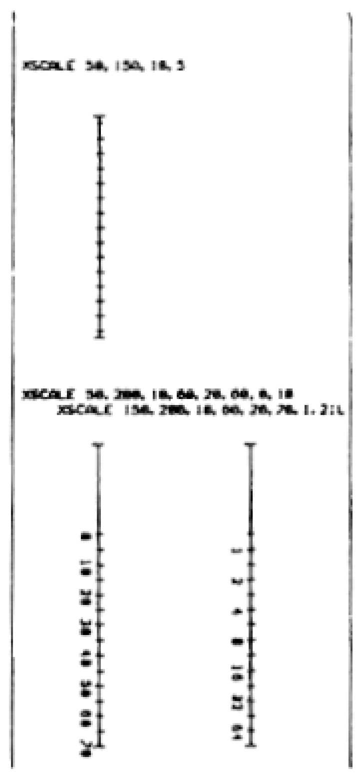
**XSCALE** o,x,dx,xz1,dw,wl1,we,sw;L
Abbreviation: **XS.**

- o: y-coordinate of the lower line of the x-scale (-1999 ≤ o ≤ 1999)
- x: Units in the x-direction (1 ≤ x ≤ 1999)
- dx: distance between two subdivisions (0 < dx < 65535)
- x1: Distance of the 1st subdivision from the left edge (0 ≤ xl ≤ 65535)
- dw: Distance between two labels (0 ≤ dw ≤ 65535)
- w1: Distance of the 1st label from the left edge (0 ≤ w1 ≤ 65535)
- we: value for the left margin
- sw: step size (or multiplication factor for ;L)

Draw an X-scale with subdivisions and labels. The numbers are centered at the determined X coordinates and 12 plotter units below the scale. The labels are placed parallel to the X-axis with **CSIZE 1**. If the **USING** format is exceeded, no output is performed.

When selecting the parameters for dw and the **USING** format, ensure that there is sufficient space between the outputs, otherwise the numbers will be written on top of each other. An offset allows the entire scale to be shifted in the positive Y direction. If no labeling is required, the abbreviated command can be used.

Example:



Syntax and description:

**XSCALE** o, x, dx, x1.

Draw the x-scale with a double line and two longer boundaries at both ends. Starting at position x1, small divisions are drawn with an increment of dx. If x1 exceeds the value of x, further divisions are omitted. If dx is specified as 0, only the division at x1 is output.

**XSCALE** o,x,dx,x1,dw,w1,we,sw
**XSCALE** o,x,dx,x1,dw,w1,we,sw;L

The determined numerical values are output centered 12 plotter units below the scale. The specified value we is output at position wl, and the other values are output at a distance of dw. The value dw is added to the numerical value we for the next numerical output. Negative values can also be used. If a ;L is appended to the command, multiplication occurs instead of addition. This allows for a logarithmic scale. Negative values or 0 for sw are also permitted here. However, this leads to nonsensical scale labels.

**YSCALE** o,y,dy,y1,dw,w1,we,dw;L
Abbreviation: **YS**.

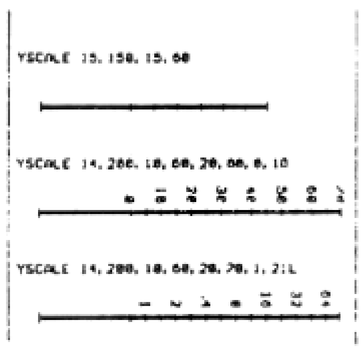
y: units in the y-direction (1 ≤ y; y+o ≤ 1999)
dy: distance between two subdivisions (0 ≤ dy ≤ 65535)
y1: Distance of the 1st subdivision from the bottom edge (0 ≤ y1 ≤ 65535)

Parameters otherwise as **XSCALE**

Draw a Y-scale with subdivisions and labels. The numbers are displayed right-aligned to the left of the scale. The labels are perpendicular to the Y-axis with **CSIZE 1**.

All parameters have the same meaning as with the **XSCALE** command, except the labels start at the bottom and all values refer to the Y axis. An offset moves the entire scale in the Y direction, but does not change its size.

Syntax:



YSCALE o,y,dy,y1
YSCALE o,y,dy,y1,dw,w1,we,dw
YSCALE o,y,dy,y1,dw,w1,we , dw ;L

**DRAW** b,a,e,wa,we,0,X,y
Abbreviation: **DR.**

b: number of the buffer (1 b **BUFNUM**)
a: Number of the first point (1 a **BUFLEN**)
e: Number of the last point (1 e **BULEN**)

wa: Voltage value at the bottom of the output field
we: Voltage value at the top of the output field
o: Coordinate of the lower left edge (-1999 o 1999)
x: Units in the X direction for the time axis (1 x 1999)
y: Units in the Y direction for the voltage range (1 y, y+o 1999)

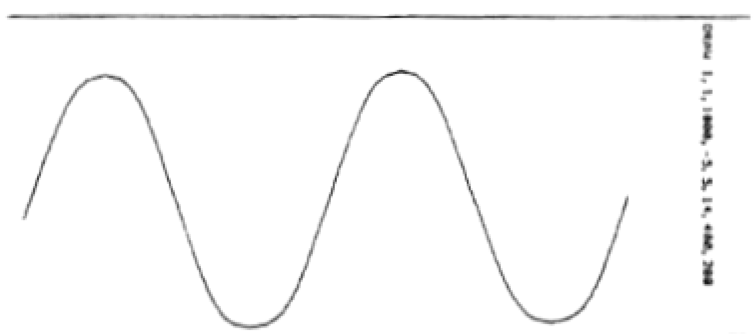
Output of the measured values stored in a buffer of the MC-12(A) or MC-10(A). The voltage range to be output can be freely selected independently of the **BUFRANGE**.

If the units in the X-direction are not equal to the number of measured values to be output, linear interpolation is performed between these measuring points. By selecting a small voltage range to be displayed, the resolution of the measuring system can be easily achieved. The digital gradation of the stored voltage values is not interpolated by the **DRAW** command to prevent misinterpretation of the measurement results.

If the voltage values lie outside the specified range, the line is drawn at the upper or lower limit. If we<wa is selected, the graph is also inverted accordingly.

Please note that the output of the measured values begins at position x=0. Therefore, if 256 measured values are to be output without interpolation, a length of x=255 must be selected. For the Y-axis, the visible range is limited to 216 units.

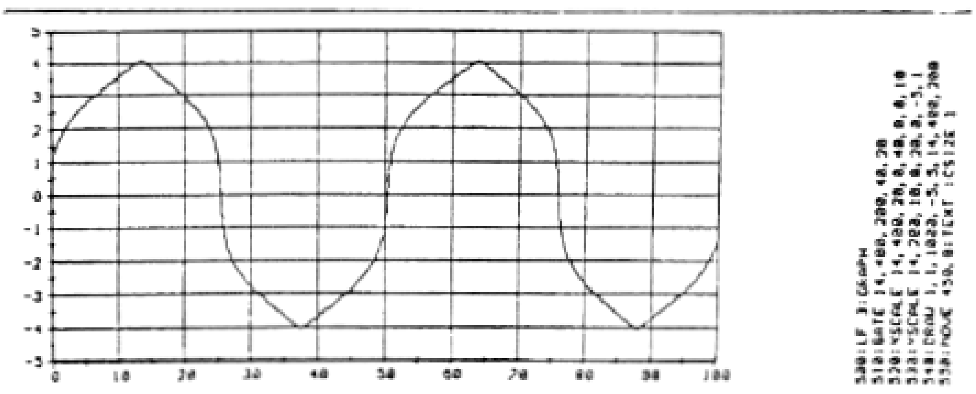
Syntax: **DRAW** b,a,e,wa,we,o,x,y



**Examples of combining commands**

The following parameters have proven effective for graphic output:
- Offset o=12 (X-axis labels are then fully visible)
- Height y=200 (Y-axis still fully visible)

If the length x > 512, there is a risk of lateral displacement when the paper is rolled back.




Error messages

**ERROR 1**: Incorrect input of parameters or commands
**ERROR 19**: Numerical value is outside the permissible range (e.g. negative values for dx, dy ...)
**ERROR 70**: Valid coordinate range from -2048 to +2047 has been exceeded or will be exceeded by the command.
**ERROR 73**: The instruction cannot be executed in the selected **TEXT** mode (enter the **GRAPH** command).

# MC12.S5V.24
# PLOTTER PROGRAM MODULE FOR
# DIN A4 PLOTTER CE 516P

Program description

This extension of the MC-12(A).S5 plotter program module offers the possibility of graphic output via the DIN A4 plotter CE-516P.

The commands for output on the A4 plotter differ in their syntax only by a leading "**P**" (e.g., **PGATE**). The meaning and order of the required parameters are identical to the standard plotter commands.

For the usable paper format and the corresponding drawing areas, please refer to the plotter description on page 64.

To connect the plotter to the MC-12(A), use the supplied cable.

The setting switches on the back of the plotter must have the following positions


In addition to the standard plotter commands, some commands have been included to make working with the CE-516P plotter easier.

### Command description

**PGINIT**
The V.24 interface of the MC-12(A) is enabled and the transmission parameters are set correctly. The CE-516P plotter is put into graphics mode.

**GRAPH**
The CE-516P plotter is switched to graphics mode. This command is only necessary if the plotter was switched to text mode after **PGINIT**.

**PTEXT**
The plotter switches to text mode. Any text can be output using the **PRINT#-232, "Sample Text"** command.

**PCOLOR**
The plotter changes the color.
Syntax: **PCOLOR** n
n: Color number (0...3)

**PSORGN**
The current pin position becomes the new origin point.

When working with the plotter, please note that after switching on and initializing with **PGINIT**, a maximum movement of 1 cm in the positive y-direction is possible. Therefore, it is recommended to position the paper with the **PMOVE** o command so that the lower left corner of the graphic to be created is under the pen (e.g., **PMOVE 150,-1000 : PSORGN**). The **PSORGN** command defines this point as the origin point for the graphic.

**All commands at a glance**

| | |
|---|---|
| **PGATE** | Offset,X,Y ('DX,DY ('x1,Y1)) |
| **PMOVE** | X,Y |
| **PXSCALE** | Offset,X,Y (,DW,W1,WE,SW GL) |
| **PYSCALE** | Offset,X,Y (,DW,W1,WE,SW GL) |
| **PDRAW** | Buffer,Start,End, WA,WE,Offset,X, Y |
| **PGINITE** | |
| **PGRAPH** | |
| **PTEXT** | |
| **PCOLOR** | number |

**Example Program**

The following example program first sets up two double-precision measuring buffers in the MC-12(A) and describes them with a sine and a cosine function (takes about 2 minutes).

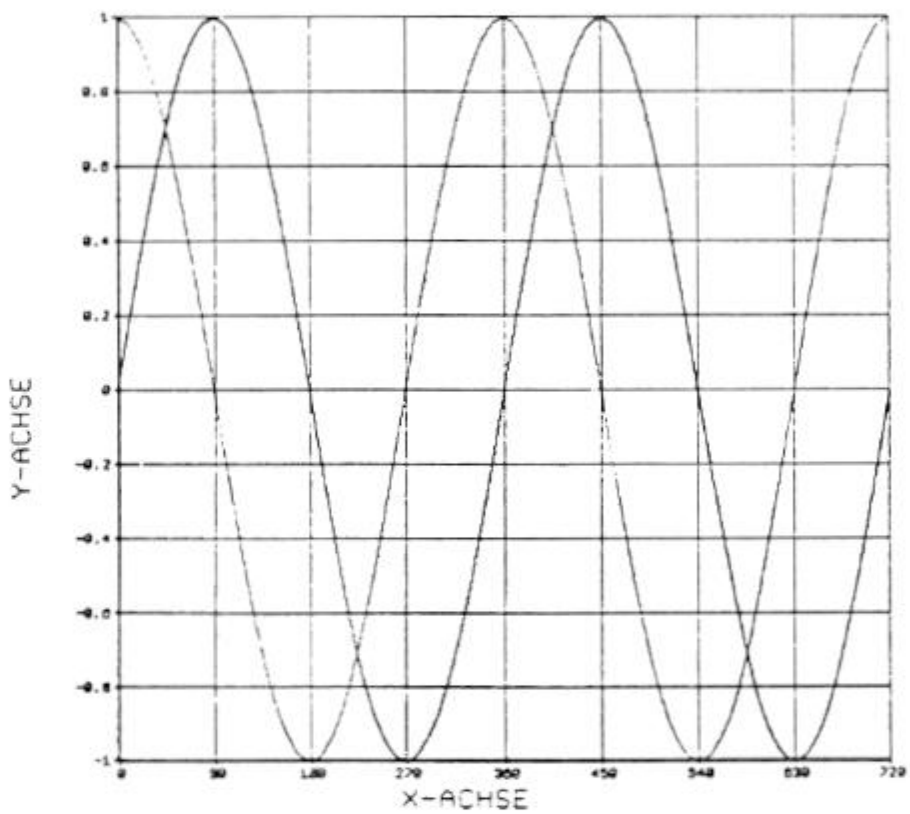
These two functions are then plotted in a coordinate system.

Starting in line 120, the two axes are labeled X-AXIS and Y-AXIS. Since no special commands are provided for this, the plotter must be addressed using the command "**PRINT#-232**." Further information can be found in the plotter's manual.

```
10: "A":DBUFINITE 2.2
20: LOAD BUFFER 1.5IN(360/256*POSITION)
30;: LOADBUFFER 2,C0S(360/256*POSITION)
40: "B":PGINIT :PMOVE 150,-700:PSORGN:PCOLOR 0
50: PGATE 12,512,500,64,50
60: PXSCALE 12,512,64,0.64,0,0.90
70: PYSCALE 12,500,50,0,50,0,-1,.2
80: PCOLOR 1
90: PDRAW 1,1,512,-1,1,12,51 2,500
100: PCOLOR 3
110: PDRAW 2,1,512,-1,1,12,512,500
120: PCOLOR 0:PRINT#-232,CHR$(27)+"?2": PRINT#-232,CHR$(27)+"c12"
125: PMOVE 190,-20:PRINT#-232,"P"+"X-AXIS”
130: PMOVE -60,190:PRINT #-232,"Q3":PRINT#-232,"P"+"Y-AXIS”
140: PMOVE -150,-500:PSORGN :PGINIT :END
```

**Error messages**

**`ERROR 68`**: Incorrect parameters or **`COM OFF`** (enter **`PGINIT`**)
**`ERROR 69`**: Printer not connected, not turned on or wrong cable



# MC 12.2
# Pt 100 Measuring Amplifier

**HOW IT WORKS**

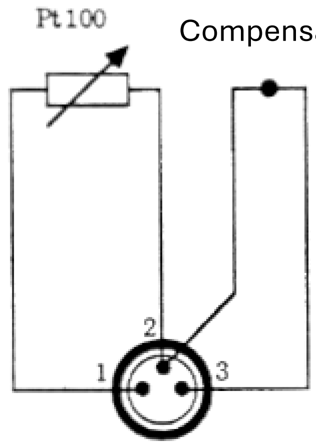
For temperature measurements, each of the five input channels can be equipped with a Pt 100 measurement amplifier module. The module consists of a highly accurate, low-noise chopper amplifier with an offset drift of approximately 1µV. The measurement amplifier operates as a bridge amplifier using four-wire technology.

The relationship between resistance and temperature is non-linear. The basic values of the Pt 100 measuring resistor are defined in DIN 43 760. Using the specified calibration curve, the bridge voltages measured with channel K (K=1...5) are directly converted into temperature values (in °C) by the MC-12 system and displayed:

v = A+B*CHA(K) + C*CHA(K)"^2 + D*CHA(K)'^3

with the coefficients

| | **A** | **B** | **C** | **D** | **Error** |
|---|---|---|---|---|---|
| Pt 100<br>-100 °C...270 °C | 0.0 | 53.0 | 0.717 | 0.0 | ±0.4C |
| Pt 100<br>-100 °C...5450 °C | 0.0 | 106.5 | 2.94 | 0.0 | ±2.0C |
| Pt 100<br>-100 °C...800 °C | 0.0 | 211.2 | 11.37 | 0.68 | ±0.7C |




Plug from above

**EXAMPLE**

Measuring amplifier on channel 4; temperature range 0 °C...200 °C: Connect the sensor to input socket 4 (Figure 1). Enter the following commands via the PC-1500(A):

- **`SET FUNCTION 4.53*CHA4 + .717*CHA4^2`** (Linearization)
- **`USING "####.#"`** (Set display format)
- **`MULTIMETER`** (Turn on multimeter)
- <4> (Select channel 4)
- <DEF> key (Switch from volts to degrees)

The transfer function is deleted by the **`INIT`** command or by overwriting it with another function.

If the temperature values are to be accessed directly in a BASIC program, the **`INFUNKTION`** command must be used (see manual MC-12 chapter 7.3).

Working is made considerably easier if the commands **`SETFUNCTION 4, 53`**.... and **`USING"#`**... are assigned to the reserve keys of the PC-1500(A) (see PC-1500(A) manual page 97).

## TECHNICAL DATA

| Type | Measuring range | Accuracy | Resolution |
|---|---|---|---|
| MC12.2a | -200 °C...+270 °C | ±0.2K | 1 LSB |
| MC12.2b | -200 °C...+540 °C | ±0.4K | 1 LSB |
| MC12.2c | -200 °C...+2800 °C | ±0.8K | 1 LSB |

## INSTALLATION OF THE MODULE

The input channel assembly is located on the rear of the MC-12(A).

If the MC-12.2 is to be installed later, the MC-12(A) must be opened (3 screws each on the front and back of the MC-12(A)). Four 7-pin socket connectors B1...B4 are located on the lower circuit board. Each of these sockets can accommodate a measuring amplifier. Plug the module into a free slot B1...B4 (Figure 2).

If socket B5 on the upper board is to be used for an input module, e.g. if sockets B1 to B4 are already occupied, a special module with an angled connector strip is required (please specify when ordering).

MC-12(A)

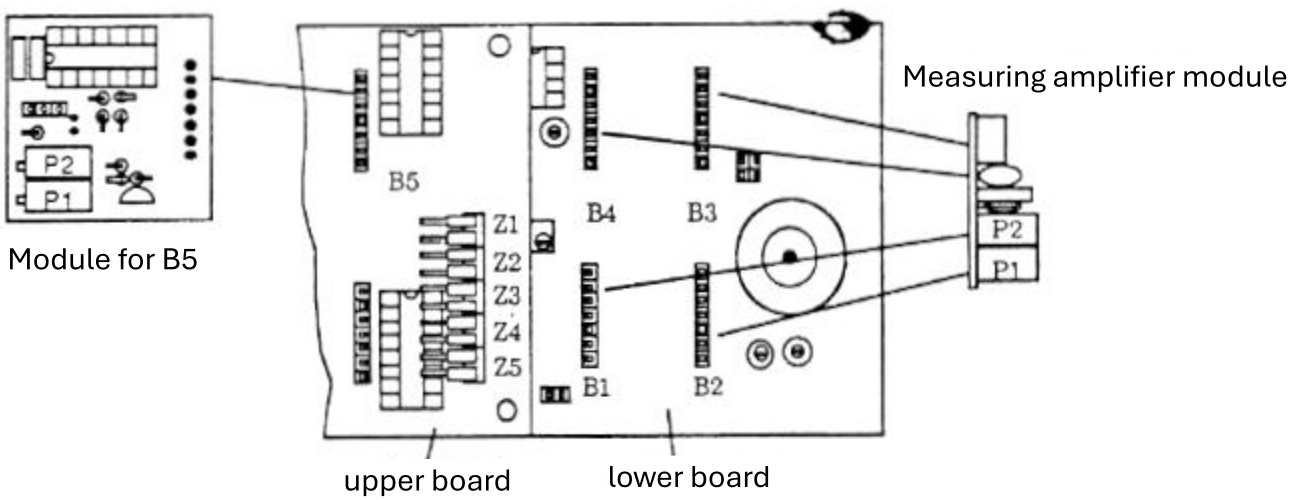



Image 2

On the upper circuit board, there is a 10-pin female connector to which the signal lines from the input sockets Z1...25 are plugged. Disconnect the connector of the channel intended for the resistance measuring amplifier and plug it into the measuring amplifier, as shown in Figure 3. The connecting cable coming from the measuring amplifier is inserted into the vacant position on the 10-pin connector.

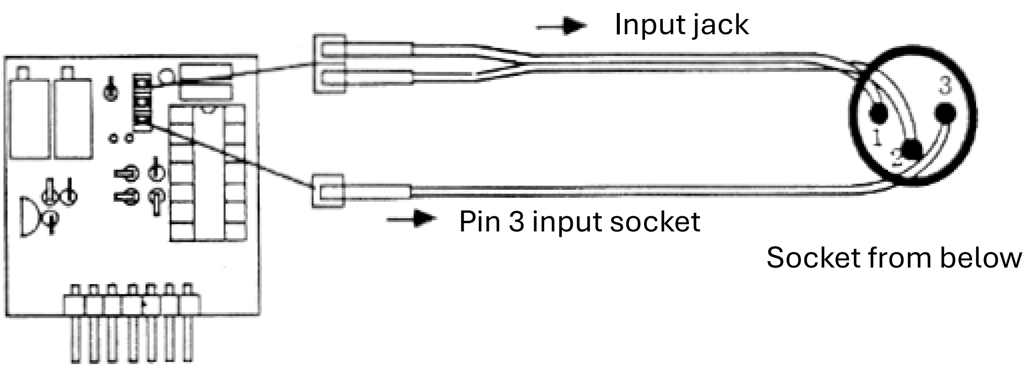



Image 3

To operate in four-wire technology, the single-wire cable must be soldered to PIN 3 of the 3-pin input channel socket. Cut the cable to the appropriate length beforehand. The plug of this cable is inserted into the free space on the 3-pin socket header of the measuring amplifier (see Figure 3).

If the amplifier is to be operated in two-wire technology, the additional single-wire cable can be omitted. However, a solder bridge is required between socket contacts 2 and 3 on the underside of the amplifier board (see Figure 4).

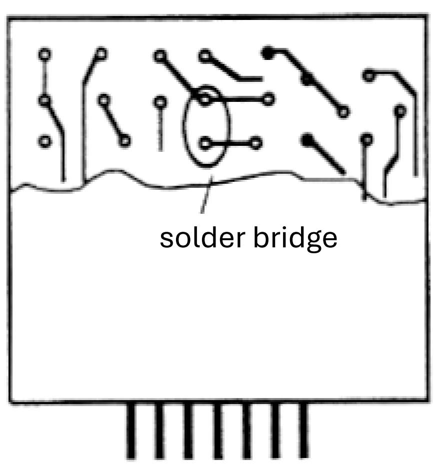



Image 4

## ADJUSTING THE MEASURING AMPLIFIER

The measuring amplifiers are factory-adjusted for four-wire technology. Readjustment of the measuring amplifier is best performed using a calorimeter vessel (thermos flask). Place a few ice cubes (distilled water) into the vessel and pour in a little water. After approximately 15 minutes, the ice point will be reached.

There are two adjustment options:

### ADJUSTMENT OF THE MEASURING AMPLIFIER BY P2

Place the sensor in the ice water. Turn on the MC-12(A) and enter multimeter mode (**`MULT.`**). Select auto-range mode (A) and select the Pt 100 measurement channel (e.g., 4). Use **P2** to set the display value to 0V (1mV = 0.05 °C for module 12.2a).

### ADJUSTMENT BY CHANGING THE LINEARIZATION FUNCTION

If readjustment is required later, this can be done without opening the MC-12(A). Place the measuring sensor in the ice water. Switch on the MC-12(A) and enter the linearization function. Call up the multimeter operating mode (MULT.) with auto-range operation (A). Select the Pt 100 measuring channel (e.g. 4) and use the <DEF >key to switch the display from volts to degrees Celsius. Note the displayed temperature value. This value is now subtracted from the coefficient A of your linearization function. Enter the modified linearization function and check the zero point again.

# MC 12.3
# RESISTANCE MEASURING AMPLIFIER

**How it works**

For measurement, each of the five input channels can be equipped with a resistance measurement amplifier module. The module essentially consists of a temperature-stabilized constant current source powered by the MC-12 (A).

The resistance to be measured is passed through a constant current of 1uA or 1004A, depending on the type of input amplifier. The voltage drop across the resistance to be measured is measured by the MC-12 (A).

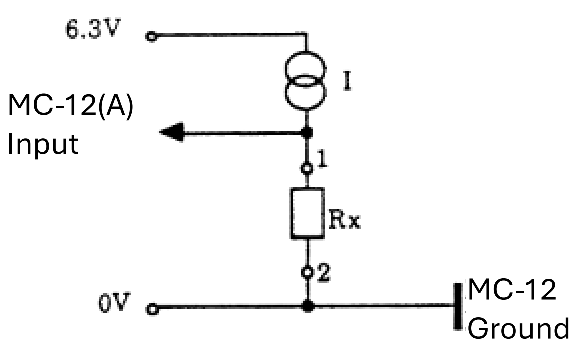
The voltage value U shown on the display is proportional to the resistance value R.

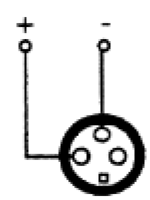
| | | |
|---|---|---|
| MC-12.3a | U=R/10,000 | **SETFUNKTION 4,CHA4*1E4** |
| MC-12.3b | U=R/10,000,000 | **SETFUNKTION 4,CHA4*1E6** |

To directly display the resistance values, the **SETFUNCTION** command can be used,

e.g. **SETFUNKTION 4,CHA4*1E4**

If you now press the "DEF" button in multimeter mode, the converted value will be displayed.




Socket on MC-12

## Technical data

| Type | Measuring range | Resolution | Accuracy |
|---|---|---|---|
| MC12.3a | 500Ω - 50kΩ | min. 0.4Ω | ±1% (2LSB) |
| MC12.3b | 5kΩ – 5MΩ | min. 40Ω | ±1% (2LSB) |

## Installing the module

**MC-12(A)**

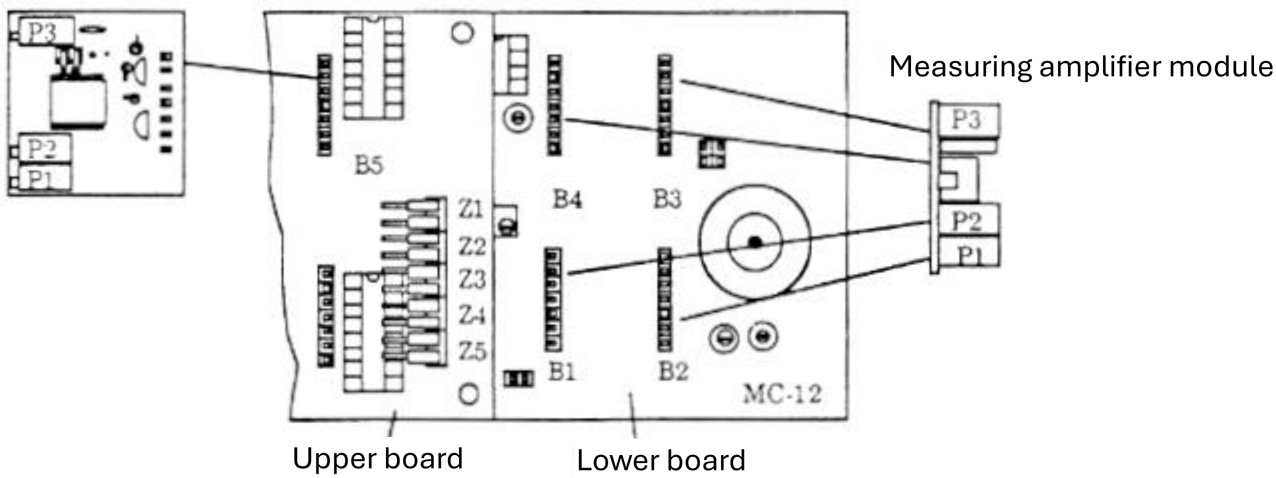



Image 2

To install the MC-12.3, the MC-12 (A) must be opened by removing the four (6) side screws. Four 7-pin socket connectors are located on the lower circuit board. Each of these sockets can accommodate a measuring amplifier. Plug the module into a free slot B1...B4 (Figure 2).

On the upper circuit board there is a 10-pin socket strip to which the signal lines from the input sockets E1...E5 are plugged. Pull out the plug of the channel intended for the resistance measuring amplifier and plug it onto the measuring amplifier, as shown in Figure 3.

The connecting cable coming from the measuring amplifier is placed in the freed position on the 10-pin connector strip.

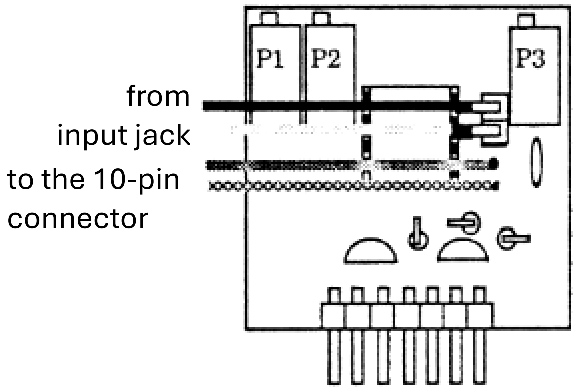



Image 3

**Adjustment of the measuring amplifier**

The measuring amplifiers are factory-adjusted. If adjustment is required, proceed as follows:

**Zero-point adjustment**

Short-circuit the measuring channel; turn on the MC-12 (A) and enter multimeter mode (**`MULT.`**). Enter auto-range mode (A) and select the resistance measuring channel (e.g., 4). Use **P2** to set the display value to 0V.

**Gain adjustment**

Connect a resistor of known value to the measuring channel. The displayed value can be corrected using **P3**.

# MC 12.5
# Measuring amplifier for thermocouples

**General description**

The thermocouple amplifier consists of a high-precision, low-noise chopper amplifier with an offset drift of approximately 0.1 uV/°C. The amplifier does not have a compensation measuring point.

**Position of the module**

If the measuring amplifier was ordered together with the MC-12(A), the amplifier is already installed in the MC-12(A) at the factory. The channel equipped with the amplifier can be found on the label on the rear of the device.

**Using the MC-12.5**

The basic values of the thermoelectric voltage series are defined in DIN 43 710. The relationship between thermoelectric voltage and temperature is non-linear. The accuracy of the measurement is influenced by recrystallization, evaporation of individual components, contamination, and mechanical stress.

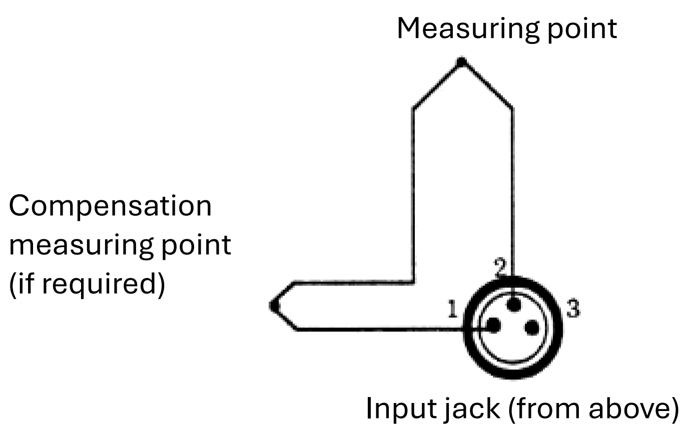



Figure 1: Connection of the thermocouple

Using the calibration curves we have developed, the thermoelectric voltages measured with channel K (K1..5) are directly converted into temperature values (in °C) by the MC-12(A) system and output:

= A + B*CHA(K) + C*CHA(K)^2 + D*CHA(K)^3

with the coefficients

| **Thermocouple** | **A** | **B** | **C** | **D** |
|---|---|---|---|---|
| **FE-Konst.** | | | | |
| -100°C...200°C | 0.0 | 199.0 | -32.49 | 15.99 |
| 200°C...600°C | 7.6 | 171.0 | 4.58 | -0.995 |
| **PtRh-Pt** | | | | |
| 0°C...200°C | 0.0 | 1760.0 | -3510.0 | 6438.0 |
| 200°C...1300°C | 33.2 | 1207.6 | -267.2 | 61.25 |
| **NiCr-Ni** | | | | |
| 0°C...500°C | 0.0 | 246.1 | 0.701 | -1.2631 |
| 400°C...900°C | 5.3 | 253.0 | -10.35 | 1.816 |

**Measurement example**

Measuring amplifier on channel 5; FE-const. sensor; temperature range 0°C to 200°C;

Connect the sensor to input socket 5 (Figure 1). Enter the following commands via the PC-1500:

**`SETFUNKTION 5,19,9*CHA5-.325*CHA5"2+ 1.59E-2*CHA5’3`**
**`USING "####.#"`** (Set ad format)
**`MULTIMETER`** (turn on multimeter)
**5** (select channel 5)
<DEF> key (switch from mV to C)

The transfer function is deleted by the **`INIT`** command or by overwriting it with another function.

If you want to access the temperature values directly in a **`BASIC`** program, you must use the **`INFUNKTION`** command (see MC-12(A) manual, chapter 7.3).

Working is made considerably easier if the command **`SETFUNCTION 5, 19,9*...`** and **`USING`**"... are placed on reserve keys of the PC 1500 (see PC 1500 manual page 97).

**General technical data**

- Amplification factor: 100V (1mV display corresponds to 10uV thermoelectric voltage)
- Drift: 10WV over the entire temperature range (0°C..40°C)
- Frequency response: 0 Hz... 10 Hz
- Input circuit: differential amplifier circuit

# MC 12.6
# Strain Gauge Amplifier

**How it works**

The measuring amplifier for strain gauge bridges features an integrated power supply for measuring bridges with an impedance greater than 350 ohms. Bridges with lower input impedances require an external power supply. The bridge supply provided by the MC-12.6 is 5 V. The quiescent current consumption of the entire measuring amplifier is a maximum of 2 mA.

The gain of the measuring amplifier can be adjusted within a range of 1 to 100. The back offset can also be compensated over a wide range. The offset noise and offset drift of the measuring amplifier are far below those of commercially available measuring bridges. The measuring amplifier is designed for full-bridge circuits. For half- or quarter-bridge circuits, the measuring bridge must be supplemented with external resistors.

By assigning a transfer function to each channel of the MC-12(A), measured values can be displayed and output in multimeter mode in the measured physical quantity. For more information, see pages 45 and 24 of the MC-12(A) manual.

**Technical data**

| | |
|---|---|
| **Bridge supply:** | 5V +0.5%<br>Bridge input impedance greater than 3500 Ω<br>Temperature drift +0.1 mV/°C |
| **Gain factor:** | ...approx. 100 (adjustable)<br>Depending on the bridge impedance |
| **Linearity:** | better than 0.1% |
| **Offset correction:** | (adjustable)<br>Drift 1µV/°C |
| **Frequency response:** | 0 - 30 kHz |

Assignment of the 5-pin output socket

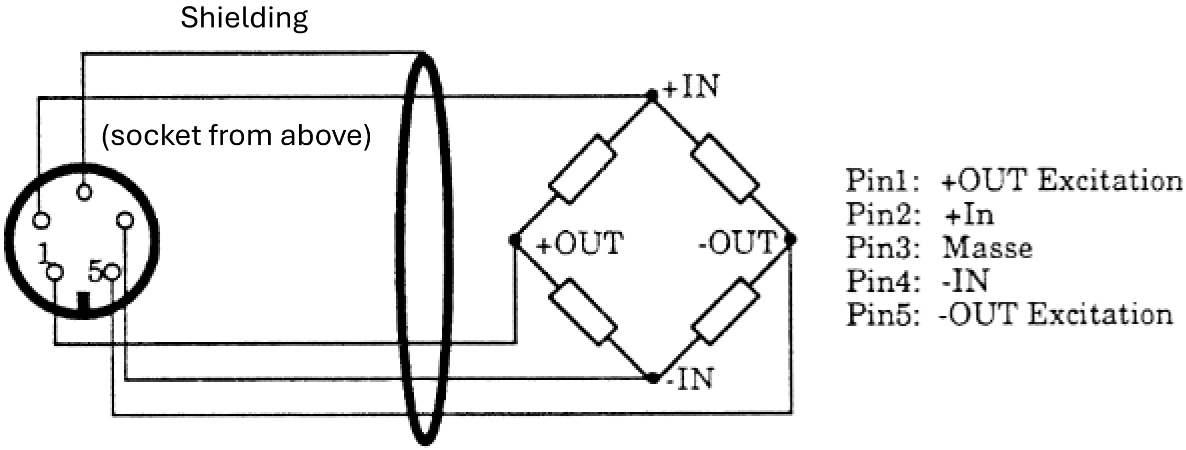



Image 1

**Adjustment of the measuring amplifier**

The measuring amplifier is factory-installed in the MC-12 (A). Before beginning the first measurements, the amplifier must be calibrated to the specific characteristics of the strain gauge bridge. Please note that the same strain gauge sensor must always be used on a strain gauge channel, as each sensor has different calibration values.

Before beginning the adjustment, solder the supplied 5-pin connector to the strain gauge sensor cable (Figure 1). If you don't have a calibration sheet for your strain gauge sensor, you'll need to use an ohmmeter to measure the sensor's output resistance between pins 2 and 4. It's also important to have as accurate information as possible about the sensitivity of the strain gauge sensor. This information can generally be found in the manufacturer's calibration report for each sensor. For the adjustment, you'll need a 3.5-volt digital multimeter and an adjustable voltage source capable of generating voltages up to 4V.

To begin the adjustment, first loosen the six side screws on the MC-12 (A). Connect the MC-12 (A) to the CE 150 and open the MC-12 (A) cover. Turn on the measuring system via the PC-1500A and enter the following short program:

```
10: MCON
20: INPUT "SENSITIVITY(MV/xx)= ";S
30: INPUT "EXCTATION (V) = ";E
40: INPUT "SCALE (xx/V) = ";V
50: INPUT "R(out) = ";R
60: A=S/(E/5)*1E-3:B=1/AV:F=B/2"R
70: C=F/4990+1:U=4/C:USING "##.###"
80: LPRINT "CAL.VOLTAGE (4V) =";U
90: PRINT "CAL.VOLTAGE (4V) =";U
```

In the following we use as an example a strain gauge accelerometer with the following manufacturer specifications:

| | |
|---|---|
| Sensor data: | (accelerometer) |
| Sensitivity: | 0.654 mV/g |
| Excitation: | 5V |
| Rin: | 572 ohms |
| Rout: | 515 ohms |
| Range: | +250g |

Start the program in RUN mode by typing **RUN** and pressing the <ENTER> key.

| Program message | Input | Unit |
|---|---|---|
| SENSITIVITY (mV/xx) = | 0.654 | (mV/g) |
| EXCITATION (V) = | 5 | (V) |
| SCALE (xx/V) = | 50 | (g/V) |
| R(out) = | 515 | (Ohm) |
| CAL.VOLTAGE (4V) = | 1.551 | (V) |

When the program asks for **SCALE** (xx/V), the desired gain factor is specified. Since the sensor's operating range is +250g, a value of 50g per volt is reasonable. If the sensor's operating range is not to be fully utilized, a different value can be entered here. For example, if 20g per volt is entered, the resolution is 2.5 times higher, but the operating range is limited to +100g (100g then corresponds to a reading of 5V).

Set the calibration voltage of 1.551 V calculated by the program on your adjustable voltage source. This voltage must be connected between pin 5 (-OUT) and pin 2 (+ IN) of the 5-pin input socket.

Turn the offset correction potentiometer P2 (Figure 2) counterclockwise until it stops (a quiet click is heard).

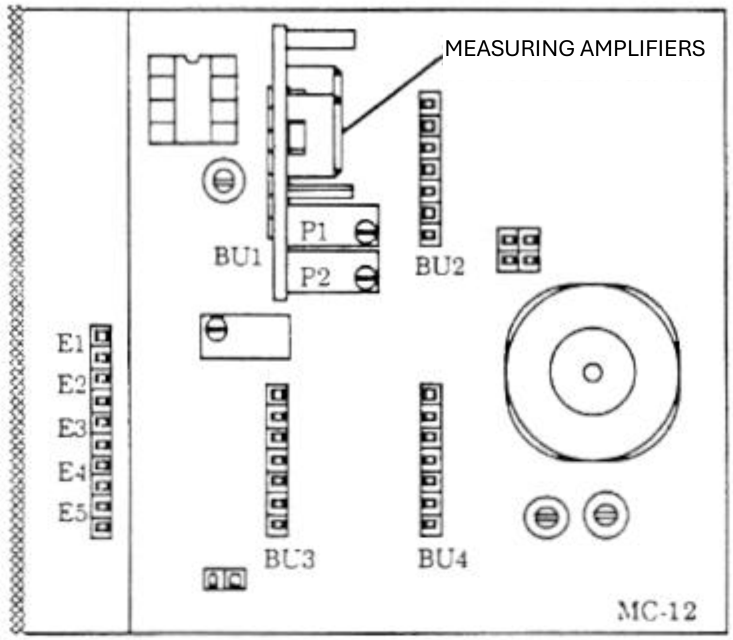



Image 2

Press the <BREAK> key to exit the program and enter the **MULT.ENTER** command to return to multimeter mode. Select the strain gauge channel by entering the channel number and switch the multimeter to normal mode by pressing the N key. Use the f key to ensure that the largest measuring range is selected.

If the voltage indicator is in the right **OVERFLOW** range, turn the gain potentiometer P1 counterclockwise; if the voltage indicator is in the left **OVERFLOW** range, turn the gain potentiometer P1 clockwise. Adjust the gain so that the display jumps between 3.99V and 4.03V.

Now connect the strain gauge sensor. Turn the offset potentiometer **P2** clockwise to set the offset to 0V. Press button A to switch to **AUTORANGE**. The offset can then be set very precisely.

In the example chosen, the gain factor is approximately 30. This means that 1 g corresponds to 20 mV. If the offset were set to +20 mV, the error relative to the sensor's maximum range would be 0.5%. In fact, however, the offset can be set precisely to +1 mV in this example. Please note that even temperature-compensated strain gauge bridges generally exhibit an offset drift of approximately 0.1 - 0.2% per 10 °C. It is therefore important that the sensor is thermally compensated and has been connected to the MC-12 (A) for some time with the operating voltage switched on. This prevents errors that can arise from self-heating.

If you have the option of checking the sensor's gain factor using a known load, you can perform a fine adjustment using the gain potentiometer P1. Since adjusting the gain factor also changes the offset, differential measurements must be performed. First, measure the bridge voltage with no load and then with a load. If the voltage difference is too large, turn potentiometer P1 counterclockwise; if the difference is too small, turn potentiometer P1 clockwise. Once the gain factor is correctly set, readjust the offset.

Since some manufacturers provide different information for sensitivity in their data sheets, here is another example:

Sensor data: (Pressure sensor)

| | |
|---|---|
| Sensitivity: | 3.3 mV/V at 1000 bar |
| Excitation: | 10V |
| Rout: | 351.8 ohms |
| Range: | 1000 bar |

The sensor specified here is designed for an operating voltage of 10V. When supplied with 5V, the sensitivity is reduced by half. This is taken into account by our adjustment program.

The sensor's sensitivity is defined as 3.3 mV per volt of supply voltage at 1000 bar. At the suggested supply voltage of 10 V, this results in an output voltage of 33 mV/1000 bar.

The following entries must be made in our program:

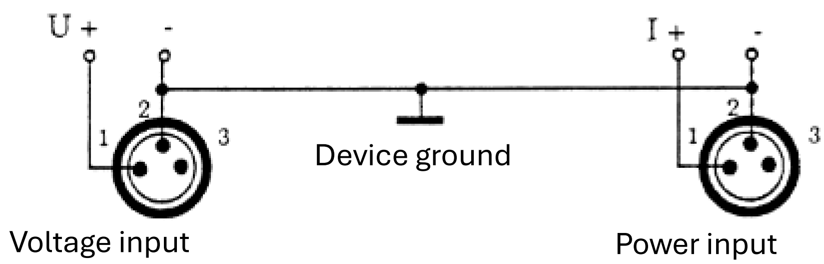
| Program message | Input | Unit |
|---|---|---|
| SENSITIVITY (mV/xx) = | 33 | (mV/1000 bar) |
| EXCITATION (V) = | 10 | (V) |
| SCALE (xx/V) = | 0.2 | (0.2 * 1000 bar/V) |
| R(out) = | 351.8 | (Ohm) |
| CAL.VOLTAGE (4V) = | 0.342 | (V) |

The adjustment process is carried out as before, except that now a voltage of 0.342 V must be applied between pin 5 and pin 2.

# MC 12.7
# Current Measuring Amplifier

How it works

The current measuring amplifier has a constant internal resistance of 1 ohm. A reading of 1 V therefore corresponds to a current of 1 A. The DEF button suppresses the V unit in multimeter mode. The current measuring section is not galvanically isolated from the other inputs. When using current and voltage inputs simultaneously, ensure that the inputs share a common ground. The channel equipped with the current measuring amplifier can be determined on the rear panel of the MC-12(A).



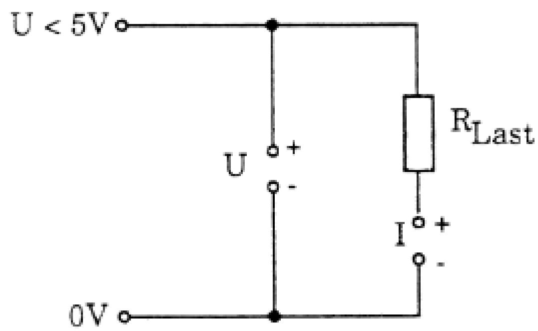

Sockets from above

**Technical Specifications**

| | |
|---|---|
| Measuring range: | 5mA...1A |
| Resolution: | minimum 351uA |
| Accuracy: | better than +150uA or 1LSB |
| Internal resistance: | 1Ω |

**Measurement example**

The power of a consumer is to be measured. The MC-12 (A) used in this example is equipped with a voltage input on channel 1 and a current input on channel 3.




Measurement setup

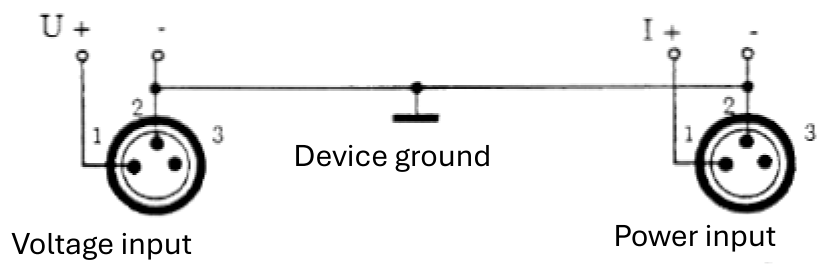
The SET FUNCTION allows a direct display of the power in multimeter mode. The following inputs must be made:

- **`MCON`** Turn on the MC-12
- **`SETFUNKTION 1, CHA1 * CHA3`** function is assigned to channel 1
- **`MULTIMETER`** the voltage on channel 1 is displayed
- Press the <DEF> button the power is displayed

# MC 12.7a
# Current Measuring Amplifier

**How it works**

The current measuring amplifier has a constant internal resistance of 100 ohms. A reading of 1V therefore corresponds to a current of 10 mA. The DEF button suppresses the V unit in multimeter mode. The current measuring section is not galvanically isolated from the other inputs. When using current and voltage inputs simultaneously, ensure that the inputs are grounded. The channel equipped with the current measuring amplifier can be determined on the rear panel of the MC-12(A).




Sockets from above

**Technical Specifications**

| | |
|---|---|
| Measuring range: | 50 mA |
| Resolution: | 2 uA or 1 LSB |
| Accuracy: | +20 uA (0.1%) |
| Internal resistance: | 100 Ω |

# MC 12.11
# 12 V ADAPTER

## How it works

The MC-12.11 adapter regulates any DC voltage in the range of 11...35V to approximately 9V. When operating with rectified AC voltage, appropriate RC elements must be used to ensure that the minimum permissible input voltage of 11V is not undercut.

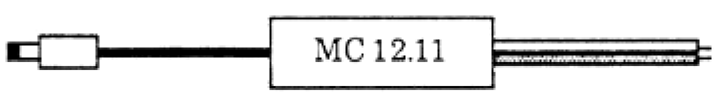
## Technical Specifications

| | |
|---|---|
| Input: | 11V...35V |
| Output: | approx. 9V, 0.5A |
| Temperature: | 70°C (OC) |
| Dimensions: | 60 x 30 x 30 mm |
| Connection: | 2-pin connector (output); 2-wire cable (input) |

## Connection

The adapter comes with a plug which is used to connect the adapter to the power source.

9V for
CE-150

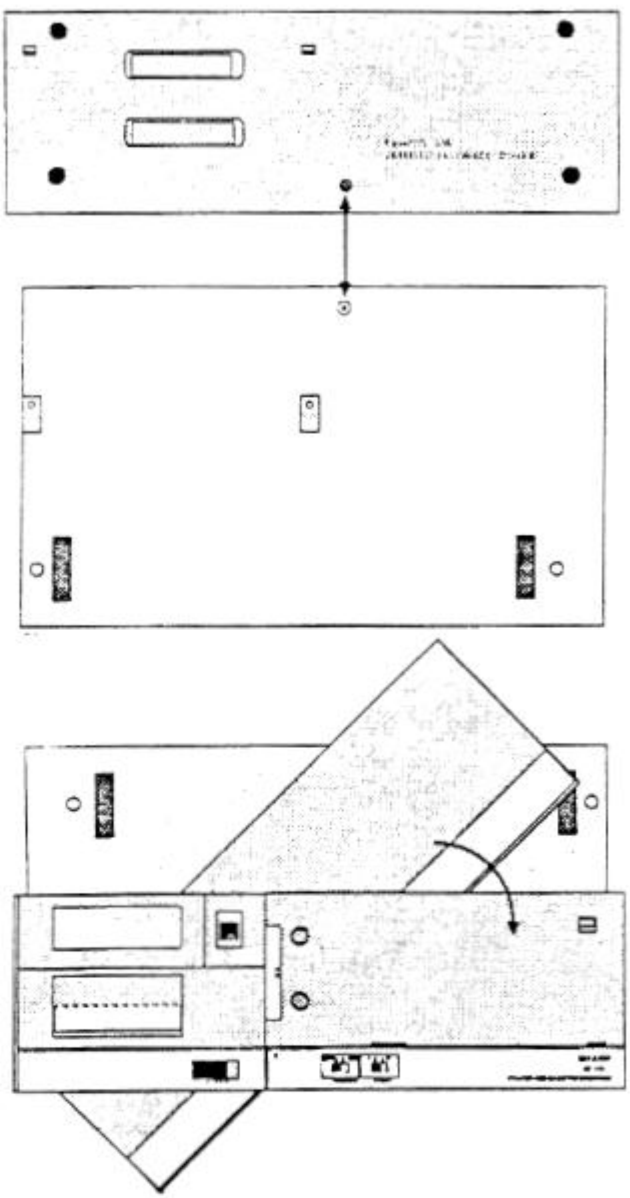



11-40 V
voltage source

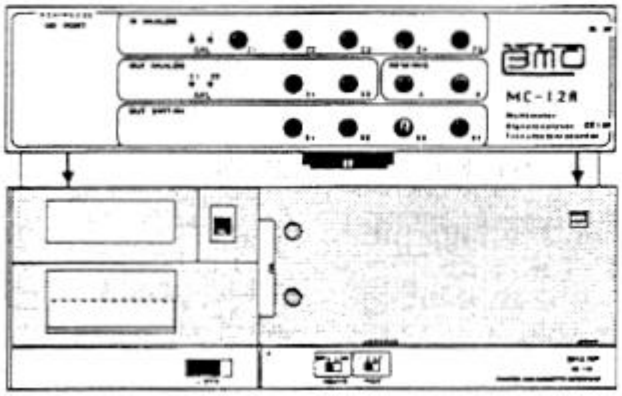
# MC 12.12
# Assembling the MC-12(A) SYSTEM

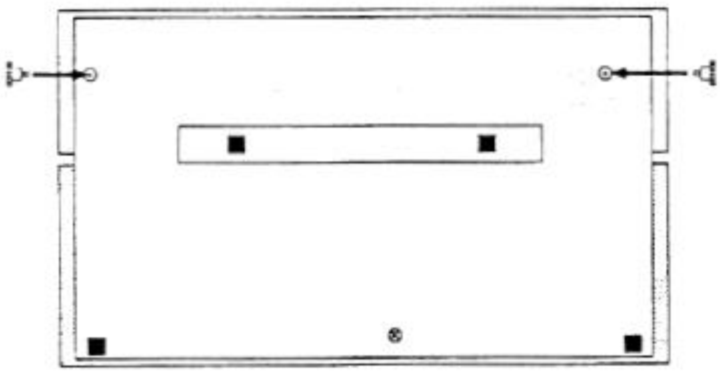
**Use an angled mounting plate**

(The straight mounting plate used on older devices cannot be used.)



Insert the CE-150 / MC-150 into the guide pin (10) with the mounting plate (9) turned to the left (Fig. 1 and Fig. 2). Turn clockwise until the hooks engage properly. Visually inspect.





Carefully insert the MC-12(A) (1) with the plug into the socket of the CE-150, ensuring that the threaded holes are centered (Fig. 3). Screw in the knurled screws vertically (do not tilt; the tension is desired!).

**Case assembly**

Place the case on a table as shown in Fig. 5. The hinges (8) can be pushed out (also to hang the case on the wall).

Slide the MC-12(A) system (1) forwards and to the left onto the guide plate (11) as shown in Fig. 4. Check that it is correctly seated: (1) must not be able to be moved upwards.

Apply the enclosed velour adhesive tape to the underside of the power supply (2). The feet must remain exposed. Insert tightly against (1). This prevents (1) from being pushed backward.

Apply the included velour adhesive tape to the underside of the tape recorder (3). Leave the feet and battery compartment free. Press firmly onto the prepared Velcro fastener.

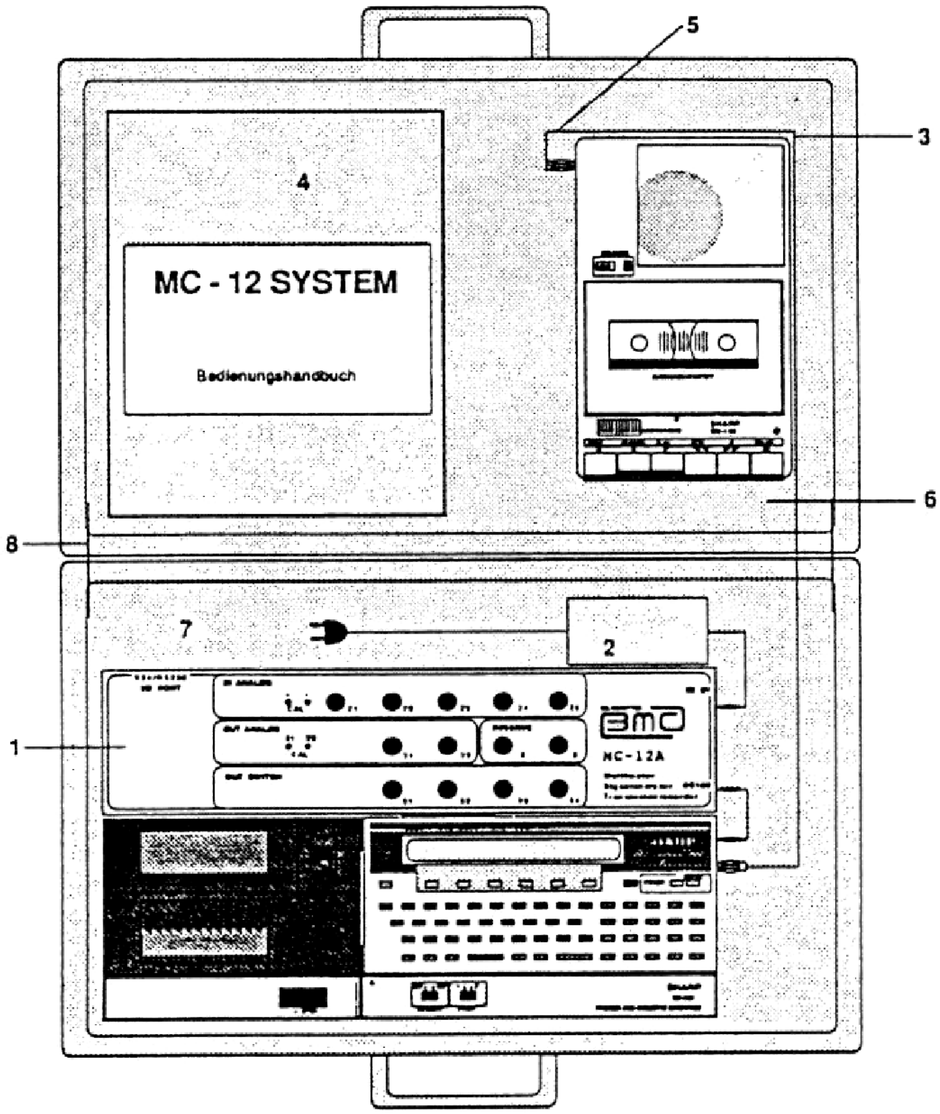
Guide the connecting cable (5) under the bevel of (3) and fasten it at (6) with Velcro,

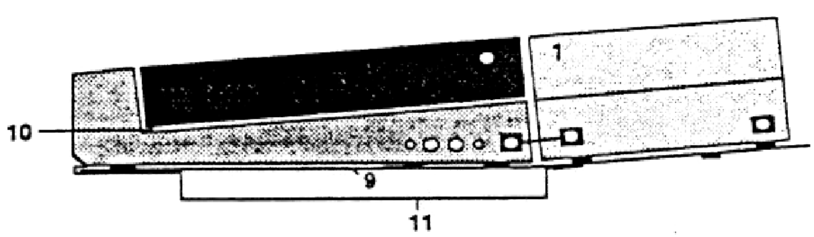
The instruction manual (4) can also be stored in the case using the included Velcro fastener, which is attached along the stitching on the back cover. Clamp the open side of the book under the rod.

Insert the cover plate (7) using the three guide pins. Slightly bend it to allow insertion on the front side of the case.

Caution: The cover plate additionally secures the system against slipping out. Therefore, do not transport it without the cover plate.

For longer transport routes or shipping, the supplied foam insert should be used for additional protection.


5
3
4
MC - 12 SYSTEM
Bedienungshandbuch
6
8
7
2
1
MC-12A


1
10
9
11

# MC 12.S2/S9

# FFT - Fast Fourier Transformation

# MC 12.S2/S9 FFT - Fast Fourier Transformation

## Table of contents

# Introduction

Fourier analysis, the representation of a time function by a sum of partial oscillations of different amplitude and phase, is the most important tool for analyzing periodic processes and for solving control processes, as well as boundary value problems in partial differential equations.

A time function y(t) with period T = 1/f1 can be expressed by a linear combination in the form

$$y(t) = a_0 + \sum_{n=1}^{\infty} [\, c * \cos(360 * f1 * n * t) + s * \sin(360 * f1 * n * t) \,]$$

For better usability in practice, the angles are given in angular lines (DEG).

The constants of the Fourier series can be calculated using the following integrals:

$$a_0 = 1/T * \int_T [\, y(t) \,]\, dt$$

$$c_n = 1/T * \int_T [\, y(t) * \cos(n * 360 * f1 * t) \,]\, dt$$

$$s_n = 1/T * \int_T [\, y(t) * \sin(n * 360 * f1 * t) \,]\, dt$$

This representation is also valid for discontinuities. At these points, an average value results. For the sufficient condition for convergence of the Fourier series, refer to the specialized literature.

After marking the time interval t(M) to t(C), the program first calculates the complex spectrum using a fast FFT algorithm, which is then converted into an amplitude-phase spectrum.

$$y(t) = A_0 + \sum_{n=1}^{128} A_n * \cos(N * 360 * f1 * t + w_n)$$

Amplitude: $A_n = \sqrt{(c_n^2 + s_n^2)}$ Phase: $w_n = \arctan(c_n / s_n)$

The fundamental frequency f1, Nyquist frequency, maximum amplitude Amax, and DC voltage component A0 are then output or transferred to the corresponding variables. This FFT algorithm considers 128 partial oscillations. Exactly 256 measured values are required for this. If fewer or more than 256 measured values are available in the selected time interval, 256 function values are automatically calculated from the available measured values using interpolation. These then form the basis for the Fourier analysis.

# CHAPTER 1

# Program Description

**Inserting the MC 12.S2 and MC 12.S9 modules**

1) Turn off the PC-1500(A) computer using the <OFF> button.
2) Insert the module into the designated compartment.
3) Assemble the computer, CE-150 plotter, and MC-12(A).
4) Turn on the computer using the <ON> button.
5) Select **PRO** mode using the **MODE** button.
6) Enter **`N E W 0 ENTER`**

The calculator is now ready. Additional Basic programs can be loaded or entered. However, there are limitations due to the modules:

- Only program labels can be found using the LIST command.
- RESERVE levels are determined by the module.
- No additional RAM/ROM module can be used simultaneously.

The calculated results of the FFT analysis occupy the first two blocks (= 512 memory locations) of the MC-12(A) measurement data memory. If a time function is stored in the first two blocks, this can also be subjected to FFT analysis. However, the signal is no longer available after the analysis.

## 1.1 Occupancy of the memory by the amplitude-phase spectrum

| **Buffer length:** | **Occupied by spectrum:** |
|---|---|
| 1 Block | Buffer 1 and 2 |
| 2 Blocks | Buffer 1 |
| More than 2 blocks | Measured value 1..512 of Buffer 1 |

## 1.2 Occupancy of the RESERVE levels

Reserve Level I: vacant

Reserve Level II:

| Key | Display | Function | Explanation |
|---|---|---|---|
| ! (F1) | **MON** | **MCON** | Turning on the MC-12(A) |
| ‘ (F2) | **OFF** | **MCOFF** | Turning off the MC-12(A) |
| # (F3) | **TRA** | **GOTO ‘ ‘** | Start the transient recorder |
| $ (F4) | **SCR** | **GOTO ‘A’** | Start the extended SCREEN mode |
| % (F5) | **MUL** | **MULTIMETER** | Calling up multimeter mode |
| & (F6) | **AUS** | **CALL &E33E** | Turning off the computer |

The <F6> key turns the computer off without initializing the printer when it is turned on. After pressing the <ON> key, you must press another (any) key to make the computer ready for input again.

Reserve Level III:

| Key | Display | Function |
|---|---|---|
| ! (F1) | **SF4** | **SETFUNCTION 4,53 * CHA4 + .717 * CHA4*2** |
| ‘ (F2) | **SF5** | **SETFUNCTION 5,53 * CHA5 + .717 * CHA52** |
| # (F3) | **US.** | **USING ' ####.#”** |

These function definitions support the use of the Pt100 at inputs Z4 and Z5 of the MC-12(A). The USING statement specifies the output format for the temperature display.

## 1.3 Assignment of program labels

Here is an alphabetical summary of all labels. Detailed descriptions of all subroutines, including the required inputs and outputs, can be found in the following chapters.

| Label | Short description | Page |
|---|---|---|
| ‘A’ | Entering the extended SCREEN mode (Main program) | 211 |
| ‘AA’ | Extended SCREEN mode as a subroutine | 203 |
| ‘B’ | Fourier analysis of a specified area | 205 |
| ‘D’ | Double integration for accelerometers | 216 |
| ‘EW’ | Individual values of the partial oscillations | 213 |
| ‘F’ | Calculating the period and frequency of a signal | 214 |
| ‘FE’ | Find the next sign change | 215 |
| ‘FO’ | Formatting a number with USING and size unit | 211 |
| ‘G’ | Output of acceleration, speed and distance | 217 |
| ‘J’ | Integration of a specified area | 217 |
| ‘K’ | Output of RMS value, distortion factor and components | 214 |
| ‘L’ | Converting a spectrum for logarithmic representation | 212 |
| ‘R’ | Effective value of an order range | 218 |
| ‘T’ | Graphical output of a Fourier spectrum | 213 |
| ‘W’ | Numerical output of frequency, amplitude and phase | 213 |

## 1.4 Additional command keys in extended SCREEN mode

| Key | Short description | Page |
|---|---|---|
| B | Fourier analysis of the marked area | 205 |
| C | Marking the 1st border | 204 |
| D | Double integration of a marked area (S2) | 208 |
| F | Output of period and frequency from cursor position (S2) | 207 |
| G | Output of acceleration, speed and distance (S2) | 209 |
| I | Re-acquisition of measured values (INSCAN) | 203 |
| J | Simple integration of a marked area (S2) | 208 |
| K | Output of RMS value, distortion factor and components | 207 |

| | | |
|---|---|---|
| L | Converting a spectrum for logarithmic representation | 205 |
| M | Marking the 1st border | 204 |
| P | Plotting the marked area | 204 |
| R | Effective value of an order range (S2) | 209 |
| S | Display of the spectrum on the oscilloscope | 206 |
| T | Graphical output of the Fourier spectrum | 206 |
| W | Numerical output of the partial oscillations of a range | 206 |

# CHAPTER 2

# ADVANCED SCREEN MODE

Start: a) Enter **RUN** or **DEF A**
b) Call label '**AA**' by other programs
c) Press **<$>** (F4) in RESERVE level II
d) Press **<Y>** when prompted **'SCREEN DISPLAY (Y/N)?'** of the extended transient recorder.

End: Press the **<E>** key

In this mode, an oscilloscope connected to the MC-12(A) outputs X1 and X2 serves as a screen, and the PC-1500's LCD display shows the current cursor position. All commands available in normal **SCREEN** mode remain valid. Refer to the MC-12(A) manual for detailed descriptions.

Note: After all outputs appearing only on the display, the program waits. Press any key to return to the corresponding subprogram or to **SCREEN** mode.

The individual commands are not always independent of one another. For example, before performing a Fourier analysis or integration, a range must be marked with the **<M>** and **<C>** keys. The required sequence of commands can be found in the overview in Appendix B.

Key: **<I>**

**INSCAN** : Re-record a signal

After pressing the <I> key, the question **'CHANGE PARAMETER (Y/N)?'** appears. If you answer **N**, a signal can be acquired using the same settings as before the program was called. If you answer **Y** (Yes) to the question, the most important sampling parameters can be reset. Only one channel is possible.

- 'SELECT: x , y' : Input of channel number x and buffer number y
- 'SCANTIME (ms): t ' : Sampling time
- 'TRIGGER (V): v' : Trigger level

The displayed values x, y, t and v can either be confirmed with **<ENTER>** or changed accordingly. The message **‘PRESS ENTER TO START SCAN’** then appears. After pressing the **<ENTER>** key, **‘SCANNING’** is displayed and recording starts. After the signal is detected or by **BREAK**, the system returns to **SCREEN** mode. Since the entries are not checked, the **INSCAN** command may result in an error and program termination. However, the program can be restarted without losing any parameters (as long as **CLEAR** is not entered) using **GOTO ‘A’.**

Key: **<M>**

Mark the first range limit

A specific position can be accessed using the **<** and **>** keys (see manual). The **<M>** key saves this position and the buffer number. The position number is shown on the display. If the CE-150's PRINT switch is in the P position, the buffer number, position, time, and voltage value are printed, and the system returns directly to **SCREEN** mode.

Key: **<C>**

Mark the second range limit

This key only allows marking the second range limit if the cursor is in the same buffer in which the first limit was already marked with M (otherwise, the error message **‘WRONG BUFFER’** is displayed). The parameters are also output here in the **<P>** position. In addition, if the position numbers are different, the time difference **t (C-M)** and the fundamental frequency **f1 = 1/ t (C-M)** are also printed.

Subsequent commands are only permitted after both range boundaries have been marked. If both range boundaries are not marked, the message **'RANGE NOT MARKED'** is displayed.

Key: **<P>**

Plot the selected area

After pressing **<P>** the question **'PLOT MORE BUFFERS (Y/N)?'** appears on the display. If **<Y>** the query **‘BUFFER ( 1.. x) :’** is displayed, where x is the maximum number of buffers. Since simultaneous plotting of several signals is only possible if the same sampling times and history have been selected, this is checked by the program. **'WRONG BUFFER'** appears if the buffer number is incorrectly entered and **'NOT SAME SCANTIME/PREHIST'** if the parameters are not identical. As long as you have not yet entered four buffer numbers you will be returned to the question **'PLOT MORE BUFFERS (Y/N)?'**. If four numbers have already been entered (marked buffer and three additional buffers) or if you answer this question with **N**, plotting begins. Beforehand the number of the marked buffer and the two limits are output.

Key: **<B>**

FFT analysis of the selected area

During the calculations, which take approximately 20 seconds, the display shows **'COMPUTING'**. Only if the PRINT switch on the CE-150 plotter is set to P will the following values be output after the transformation:

- Buffer number and both range limits
- Nyquist frequency (= 0.5 * sampling frequency)
- Fundamental frequency f1
- Maximum amplitude Amax
- DC voltage component A0

You'll then return to **SCREEN** mode.

After a transformation, the following commands are allowed:

Key: **<L>**

Convert to a logarithmic representation

For some applications, you don't want to specify the amplitude in volts, but rather the decrease in amplitude from the maximum in decibels. The **<L>** key performs this conversion:

**L =20dB*lg (An/Amax)**

This conversion can only be performed once. A drop of up to -48 dB can still be detected. However, please note:

1. After conversion, the amplitude is output in dB for T and W.
2. The value range from -30 to -48 dB has only a low resolution.

Key: **<S>**

SPECTRUM: Displays the Fourier spectrum

Since an oscilloscope can serve as a monitor for the output, this key was defined to graphically display the spectrum. The amplitude spectrum is displayed at output X1 in a normalized manner. This means that the lower edge represents the value 0 and the upper edge represents the maximum Amax. The corresponding phase is output at output X2. Here, 0 degrees of phase shift are displayed in the center, +180 degrees at the top, and -180 degrees at the bottom. Since only 128 orders are determined, only every second point represents an order (starting with the second point). The intermediate points are each set to 0 to better separate the orders. This creates a center line in the phase display. The times and voltage values shown on the PC-1500(A) display are irrelevant. The **<S>** key can be used even if no spectrum has yet been calculated. The first two blocks of the MC-12(A) memory are then displayed.

Key: **<T>**

Graphical representation of the spectrum

After pressing the **<T>** key, the question **'PLOT PHASE (Y/N)?'** appears. Depending on the answer, the phase is also displayed in the spectrum. Before the spectrum is output, the parameters are displayed (see also section B). If the spectrum has previously been converted to a logarithmic representation, the amplitude is displayed in decibels.

Key: **<W>**

Numerical output of analysis values

After pressing the **<W>** key, the number of the smallest order to be output is queried with **'BEGIN ORDER 1..128 ?',** and the largest order to be output with **'END ORDER ...128 ?'**. If the entry is incorrect (END ORDER < BEGIN ORDER or END ORDER < 1), the query is repeated. The same parameters as in B then begin to be output, plus all desired orders:

- Order number
- Frequency of this order
- Amplitude in volts or decibels
- Phase in degrees

Key: **<K>**

Output of the distortion factor

After determining the spectrum, this button is used to output

- RMS value (Zeff)
- Total harmonic distortion (Kl) in percent
- Round-wave harmonic distortion (K2) in percent
- Cubic harmonic distortion (K3) in percent

These values are determined from the spectrum. By selecting the maximum as the amplitude, the distortion factor yields the correct value even when multiple periods of an oscillation are marked.

All of the following commands are no longer included in the MC12.S9 module.

The MC12.82 module also contains:

- Determination of period duration and frequency from sign changes
- Simple integration of a defined range
- Double integration for accelerometers
- Output of acceleration, velocity, and displacement

Key: **<F>**

Period duration and frequency calculation

This key starts the acquisition of the period starting from the cursor position in the current buffer. The first query is **'PERIODS :'**. Here you enter the number of periods to be examined. Only positive values are permitted. After the entry, the first sign change is searched for, and this value is saved. The next search follows for the next but one sign change. If no more sign changes are found, the output is '/NO PERIODS FOUND'. Otherwise, the program continues until the desired number of periods is reached.

- Position number of the first point after the sign change (Pos.)
- Corresponding time (Time)
- Period number (Cycle)
- Period duration (Per.) and
- Frequency (Freq.)

output. The process is aborted without an error message if fewer than the desired number of periods are found. When operating in unipolar mode, the voltage BUFRANGE/2 is used as the limit value instead of the zero point.

Key: **<J>**

Simple integration of the selected area

This command, like the following command **<D>**, is only possible in bipolar mode. In unipolar mode, the error message **‘NOT BIPOLAR-MODE’** is displayed. When executing the command, the buffer in which the integral is to be stored must be specified in response to the question **‘INTEGRAL TO BUFFER (1 ..)?’**. It is permitted to overwrite the signal itself with the integral. When plotting the buffer, the integral is then output twice. If an offset voltage is specified, the following question **‘AUTOMATIC OFFSET (Y/N)?’** must be answered with **N**. The program then queries for the desired voltage. If Y is answered, the program automatically calculates the arithmetic mean of all measuring points in the marked range and uses the result as the offset. This is particularly useful for periodic signals. For the following question **‘PLOT INTEGRAL (Y/N)?’**, only a numeric output is possible with **N**, and additionally a graphical output of the signal and integral function is possible with **Y**.

Key: **<D>**
Double integration for accelerometers

A command has been provided specifically for the use of accelerometers. This calculates the velocity and time function from a given periodic acceleration signal using double integration. Automatic offset compensation is performed. When marking the range, it is only necessary to ensure that (one or more) whole periods are always marked. Additional inputs are the sensitivity of the sensor (query **'SENS. (mV/(m/s^2))?’**) and the buffers in which the first and second integrals are to be stored. Here, too, the integral can overwrite the stored signal. When asking **'PLOT INTEGRAL (Y/N)?'**, the graph of the integrals can be plotted with **Y** in addition to the value output. Various parameters, offsets, and factors are output:

- Buffer number, start and end of the selected signal
- Buffer number for the 1st and 2nd integral
- FACTOR 0: Factor by which the voltage value in the signal buffer must be multiplied to obtain the acceleration.
- OFFSET 0: Offset (in m/s 2 ) that must be added to the acceleration to compensate for the sensor deviation.

- FACTOR 1: Factor by which the voltage value in the buffer of the first integral must be multiplied to obtain the velocity.

- OFFSET 1: Offset (in m/s) that must be added to the velocity.

- FACTOR 2: Factor by which the voltage value in the buffer of the second integral must be multiplied to obtain the distance.

Key: **<G>**

Output of acceleration, velocity, and distance

After the double integration, pressing the **<G>** key for the current cursor position will output acceleration, velocity, and distance. However, you must be in one of the three buffers listed above and in the integration range. The offset is also taken into account when calculating the values.

The following are output:

- Buffer number, position, and time
- Corrected acceleration value al(t) (relative to the calculated zero position)
- Corrected velocity v(t)
- Deflection from the center position s(t)

Key: **<R>**

RMS value of an order range

This key allows the rapid determination of an RMS value over a large number of orders. As with the **<B>** key, a range must first be marked; otherwise, "RANGE NOT MARKED" is output. The program queries the initial and final orders (**'RMS FROM (1..128)?'** and **'RMS TO (..128)'**). The final order must be equal to or greater than the initial order. After approximately 17 seconds, the result is shown on the PC-1500(A) display. If the PRINT switch is set to **<P>**, the value is also printed. Afterward, by entering **<Y>** for **'MORE (Y/N)'?**, the RMS value of the same signal can be calculated for other order ranges in approximately 3 seconds. This function calculates a completely new spectrum, so the **T**, **W**, **L**, and **K** keys can also be used.

# CHAPTER 3

## Calling Subroutines with BASIC

All program functions of the FFT module are structured so that they can also be used in your own BASIC program through subroutine calls. The required data is transferred between a user program and the FFT module through a series of subroutines with all necessary transfer parameters.

Except for the labels '**A**' and ' ' (SPACE), all parts are designed as subroutines and return to the calling program. It should be noted that important variables are not changed (see also Appendix A, Variable List). Which variables are changed by the subroutine is specified.

The most important variable is F. It indicates whether the called subroutine has fulfilled the desired task (exception 'FO'):

F=0: Subroutine completed without error
F=1: If the parameter was entered incorrectly or if the output is not possible (e.g., no spectrum was determined before 'T' or 'W')
(F=-1: No further zeros found for subroutine N°)

All output values are formatted. The formatting routine **'FO'** modifies the variables **A$**, **Q$**, **J**, **G1**, **GI**, **K**, **ST**, and the **USING** format. These variables are no longer specified in the other subroutines.

Label: **'FO'**

Formatting a number

Input: **J** : Numeric value

Output: **A$** : Order of magnitude symbol (k=kilo; m=milli ...)
**Q$** : **USING** format
**J** : Normalized numeric value (1..999 or unchanged)
**G1** : Factor used for normalization

Changed: **G1**, **K**, **ST**, **USING** format

Each given value **J** is converted by the subroutine so that an output in scientific format is possible.

Example: Default J=1.45e4 (e.g., length 14,500 meters)
Call the routine: GOSUB 'FO'
Result: J=14.5; A$='k'; G1=0.001; USING format Q$##.#"
Output the value: PRINT J;A$;'m' results in 14.5km

Label: **'B'**

FFT analysis of a specified range

| | | |
|---|---|---|
| Input: | MB | : Buffer number |
| | MP | : Position number of the 1st boundary |
| | NP | : Position number of the 2nd boundary |

| | | |
|---|---|---|
| Output: | PS | : Number of the first point (smaller of MP or NP) |
| | PA | : Number of points used for the FFT analysis |
| | PB | : Buffer number (=MB) |
| | NF | : Niquist frequency (in Hertz) |
| | F1 | : Fundamental frequency of the FFT analysis (in Hertz) |
| | AM | : Maximum amplitude (in Volts) |
| | A0 | : DC component (in Volts) |
| | ZE | : RMS value of the signal (in Volts) |
| | KL | : Total harmonic distortion (0..1) |
| | K2 | : Square harmonic distortion (0..1) |
| | K3 | : Cubic harmonic distortion (0..1) |

Changed: **F**, **FL=1**, **I**, **P$**

After checking the input parameters, the specified range is interpolated to 256 values, an FFT analysis is performed, and then all parameters are passed to the corresponding variables. Nothing is output to the printer.

Label: **'L'**

Conversion of a linear to a logarithmic spectrum

Changed: **F**, **FL=2**, **I**, **J**

The spectrum is converted. **'COMPUTING'** appears on the display during the calculation. Afterward, the **FL** flag is changed and the display is cleared.

Label: **'T'**

Plotting the spectrum

Input: F: Flag whether phase should be drawn (if **F=1**, phase is also output) (**A0**, **AM**, **F1**, **FL**, **NF**, **PA**, **PB** and **PS** must be set by the previously called subroutine **'B'**)

Changed: **P$**, **F**, **G2**, **I**

Depending on the given **F**, the spectrum is plotted with or without phase. The various parameters are output before the graph. If no output is possible, **F=1** is set. The printer then returns to **TEXT** mode with **CSIZE 1**.

Label: **'W'**

Output range of values

Input: WA : Number of the first order (1..128)
WE : Number of the last order (**WA**..128)
(**A0**, **AM**, **F1**, **FL**, **NF**, **PA**, **PB**, **PS** as for **'T'**)

Changed: **P$** , **AN**, **F**, **FR**, **G2**, **I**, **WI**, **DEGREE** as angular unit

If **F=0** after the subroutine, the various parameters of the FFT analysis are output first, followed by the individual orders. The following data is printed for each order:

- Order number
- Frequency
- Amplitude (as voltage or in decibels)
- Angle in degrees

Label: **'EW'**

Reading individual values

Input: **I** : Order number (1..128)
(**A0**, **AM**, **FI**, **FL**, **NF**, **PA**, **PB**, and **PS** are replaced by **'B'**)

Output: **AN** : Amplitude in volts **(FL=1**) or decibels (**FL=2**)
**FR** : Frequency of this order (Hertz)
**WI** : Angle (according to the calculator setting **DEG** or **RAD**)

Changed: additionally only **F**, since no output is given

After checking the order number (in case of error F=1), the amplitude, frequency, and angle of order I are determined and passed to the calling program. If FL=2, the spectrum was previously converted to logarithmic representation by subroutine 'L', and the amplitude is output in decibels.

Label: **'K'**

Output of RMS value, distortion factor, and proportions

Input: (**ZE**, **KL**, **K2**, **K3**, **PA**, **PB**, and **PS** are set by **'B'**)

Changed: **F**

If the Fourier analysis has been carried out previously, this subroutine will output the determined values in the same way as the **K** key in the extended **SCREEN** mode.

Label: **'F'**

Period duration and frequency

| | | |
|---|---|---|
| Input: | **NA** | : Number of periods |
| | **CB** | : Buffer number to search in |
| | **CP** | : Position number to start with |
| Output: | **F** | : Here, F has three possibilities: |
| | **F=0** | : Required number of periods found |
| | **F=1** | : Incorrect buffer number or position |
| | **F=-1** | : No or fewer periods found |
| | **I** | : Next period number |
| | **CP** | : Position number after the last sign change |

Changed: **F**, **WA**, **WE**, **XX**

Analogous to the **SCREEN** command **F**, the position number, time, time difference, and frequency of every second sign change are printed. Starting at the specified position number, **CP** points to either the end of the buffer or the last sign change found.

Label: **'FE'**

Next sign change

| | | |
|---|---|---|
| Input: | **CB** | : Buffer number |
| | **CP** | : Position number to start with |

| | | |
|---|---|---|
| Output: | **F** | : Here, **F** has an additional option: |
| | **F=0** | : Next sign change found |
| | **F=1** | : Incorrect buffer number or position |
| | **F=-1** | : No sign change found |
| | **CP** | : Position of the sign change or **BUFLEN** |
| | **ST** | : Time (seconds) of the current **CP** position |
| | **SP** | : Voltage value (volts) |

Changed: additionally only **K**, **SC** and **VG**, since no output is made

This routine searches from the current position **CP** to the next sign change. From this position, the number, time, and voltage value are then determined. For **F=-1**, no sign change was found, and **CP** was set to the end of the buffer.

Label: **'J'**
Simple integration

| | | |
|---|---|---|
| Input: | **MB** | : Signal buffer number |
| | **MP** | : Position number of the 'lower' limit |
| | **NP** | : Position number of the 'upper' limit |
| | **NB** | : Buffer number in which the integral should be stored |
| | **JF** | : Flag for offset type: |
| | **JF=1** | : No offset (corresponds to offset 0 value) |
| | **JF=3** | : Offset specified in the variable **JO** |
| | | otherwise: Automatic offset determination |
| | **JO** | : Offset voltage (in volts) |
| | **JP** | : Flag for output type: |
| | **JP=0** | : No output |
| | **JP=1** | : Only numerical output of the values (factors) |
| | Other | : Additional graphical output of the integral/signal |

| | | |
|---|---|---|
| Output: | **F** | : Integration factor by which the voltage value in the integral buffer must be multiplied (unit: seconds) |
| | **JO** | : Offset voltage used (volts) |

Modified: **CB**, **CP**, **I**, **J**, **K**, **SC**, **TI**, **VG**, **SP**, **XX**

The offset is added to the voltage values in the specified buffer. If the **RANGE** is exceeded, the maximum voltage is used. These determined values are then integrated and optimized, written to the desired buffer. The signal is only changed if **MB=NB**. The factor **JF** contains both the sampling time and the optimization factor. Therefore, values written directly to the MC-12(A) memory without a sampling time can result in the value **JF=0**. Instead of a specified offset voltage JO, the maximum is used if the **RANGE** is exceeded. If an output is desired, the buffer number with range limits, offset voltage, integration factor (**INT.FAK**.), and value of the definite integral **Y(C)-Y(M)** is output. In addition, if **JP=2**, for example, the signal and integral buffer are also graphically output, with the integral utilizing the entire voltage range.

| | |
|---|---|
| Application example: | Here, a range of 43 points (468-510) was integrated from buffer 2 to buffer 5. The automatic offset was determined to be -0.151 mV. This voltage value was added to the measured values during integration. To determine an integration value, the voltage difference between the two range limits must be multiplied by the integration factor 8.80E-0.5s. The definite integral over the 43 measurement points in this case resulted in -4.28e-0.7 Vs. |

Label: **‘D’**

Double integration for accelerometers

| | | |
|---|---|---|
| Input: | **B0** | : Buffer number of the signal |
| | **B1** | : Buffer number for the 1st integral |
| | **B2** | : Buffer number for the 2nd integral |
| | **MP** | : Position number of the 1st limit |
| | **NP** | : Position number of the 2nd limit |
| | **DO** | : Sensitivity of the accelerometer (mV/(m/s 2 )) |
| | **DP** | : Flag for the output type: |
| | **DP=0** | : No output |
| | **DP=1** | : Only numerical output of the values |
| | Other | : Additional graphical output of the integrals |
| Output: | **DA** | : Position number of the smaller limit |
| | **DE** | : Position number of the larger limit |
| | **DO** | : Factor for Buffer B0 (m/s 2/V) |
| | **D1** | : Factor for Buffer B1 (m/s/V) |
| | **D2** | : Factor for Buffer B2 (m/V) |
| | **O0** | : Offset for Buffer B0 (m/s 2) |
| | **O1** | : Offset for Buffer B1 (m/s) |

Modified: **CB**, **CP**, **I**, **J**, **JF**, **JO**, **JP**, **K**, **SC**, **TI**, **VG**, **SP**, **XX**

This subroutine, which is specifically designed for accelerometers, can be used to calculate speed and distance through integration. Since an accelerometer is rarely calibrated exactly to zero, an automatic offset correction is made for both the acceleration signal and the first integral. However, this correction only works if one or more entire periods of the acceleration signal are marked. As a result, the subroutine supplies factors with which the voltage values from buffers **B0**, **B1** and **B2** can be directly calculated back into the corresponding acceleration values, speeds or distances. For the speed and acceleration, the offset 00 or 01 must also be added. If the values for a specific position are required after a double integration, the **G** command must be used.

Label: **'G'**

Output of accelerometer values

Input: **CB** : Current buffer number (**B0**, **B1**, or **B2**)
**CP** : Current position number (**DA CP DE**)

Changed: **SC**, **VG**, **TI**, **SP**

Acceleration, velocity, and distance at the current position are calculated and printed. If an incorrect buffer number or a position number outside the integration range is specified, no output occurs and the subroutine sets **F=1**.

Label: **'R'**

RMS value of an order range

| | | |
|---|---|---|
| Input: | **MB** | : Buffer number |
| | **MP** | : Position number of the 1st limit |
| | **NP** | : Position number of the 2nd limit |
| | **WA** | : Number of the first order |
| | **WE** | : Number of the last order |
| | **FL** | : Flag for the calculation mode |
| | **FL=3** | : Different order number for the same range |
| | Other | : A new spectrum is calculated before the RMS value |
| Output: | **PA** | : Number of points subjected to FFT analysis |
| | **PB** | : Buffer number (=**MB**) |
| | **PS** | : Number of the first point (neither value of **MP** nor **NP**) |
| | **ZB** | : RMS value of the order range in volts |

Changed: additionally, only **CB**, **F**, **FL=0**, **I**, **J**, **K**, **SC**, **TI**, **SP** and **VG**

If subroutine **'R'** is called for the first time for a new range, **FL** must not be 3. The FFT analysis is then first carried out to the extent necessary to determine the RMS values. However, with the same signal, the RMS value of a different order range can be calculated more quickly afterwards. For this purpose, **FL=**3 is set. The variables **MB**, **MP**, **NP**, **PA**, **PB** and **PS** are not changed compared to the first call of **'R'**. The **SCREEN ON/OFF** mode must also not be changed between subroutines, as this destroys the pre-calculated spectrum. For normal FFT analysis and output, a new spectrum must be generated after **'R**' using subroutine **'B'**. The value contained in the variable **ZB** may deviate slightly from the effective value **ZE** of the normal FFT analysis for an order range of 1 to 128. It is also possible to determine an effective value other than 0 for a specific order (**WA=WE=**order number), even though an amplitude of 0 volts was specified for the output via subroutine **'W'** or **'EW'**. This is due to the fact that the **'R'** subroutine directly calculates the effective value of the range with greater accuracy and without intermediate calculations (calculating the amplitude/phase).

# CHAPTER 4

## Description of the Advanced Transient Recorder

This new transient recorder is included in both modules. Several enhancements have been incorporated:

- Entry label (accessible via **DEF SPACE**)
- Optional output of all setting parameters on the CE-150 printer
- Output of voltage values (maximum) after multimeter operation
- Configuration of more buffers than required for **INSCAN**
- Beginning channel-to-buffer assignment from a higher buffer number
- Entering extended **SCREEN** mode

Note: The extended transient recorder is a main program. At the beginning, all basic variables are cleared with CLEAR and the MC-12(A) buffers are initialized.

Automatic protocol:

If the PRINT switch on the CE-150 is set to P, all parameters are output to the plotter. In other position no output occurs.

Input dialog:

Since the outputs and queries are usually longer than the PC-1500(A) display, a short pause is always inserted between outputs. If a question mark appears, either Y, N, +, -, or the corresponding value must be entered. Numerical entries must be completed or confirmed with ENTER. The possible answers are shown in parentheses before the question mark.

| **Display** | **Input** | **Comment** |
|---|---|---|
| `HOW MANY`<br>`BUFFERS (1..26)?` | Number of buffers | |
| `HOW MANY`<br>`CHANNELS (1..x)?` | Number of channels | x=number of buffers (<5) |

**HOW MANY BLOCKS**
**BUFFERLENGTH (1..x)?** Number of blocks/buffers x = maximum number

**SELECT**
**FIRST BUFFER(1..x)?** Number of the first buffer to be allocated

This final query only occurs if more buffers have been configured than are required for data acquisition. The trigger channel is then assigned to the specified buffer. All channels entered afterward are assigned to the following buffers in the same order.

| Display | Input |
|---|---|
| **`SELECT TRIGGER CHANNEL (1..5)?`** | Number of trigger channels |
| **`SELECT NEXT CHANNEL (1..5)?`** | Number of the next channel |

This question is repeated until the specified number of channels is reached.

**`NOW SELECT RANGES!`**
**`DIRECT VIEW WITH`**
**`MULTIMETER (Y/N)?`**

There are two ways to set the measuring range:

**Y** - Setting via multimeter operation:

After **`EXIT WITH <E>`** is issued, **`MULTIMETER`** mode is called up. The channel can be set using the **`1-5`** keys, and the range can be set using the **UP** arrow keys. If the **PRINT** switch is set to **`<P>`** when aborting with **`<E>,`** the current buffer number **N** and the maximum values of the measured values (e.g., hold marks) are printed. This is only possible for the current channel. To obtain this output for multiple channels, **`MULTIMETER`** mode must be called up again after answering **`‘RANGE OK (Y/N)?’`** in the negative.

**N** - Direct input of the measuring range:

The measuring range is entered one after the other for the selected channels.

**`RANGE OF`**
**`CHANNEL x (0..4.88V)?`** Range limit x=channel number

After entering the desired measuring ranges, the input is checked with **`'RANGE OK (Y/N)?'`**. If you answer **N** here, you start again with the selection of the setting type (**`'NOW SELECT RANGES'`** ...).

| | | |
|---|---|---|
| **`HOW MANY BLOCKS`**<br>**`PREHISTORY (0..x)?`** | History blocks | x=Maximum |
| **`SELECT SCANTIME (ms)?`**<br>**`SCANTIME (x..)?`** | Enter the sampling time | x=Minimum |
| **`TRIGGER`**<br>**`LEVEL (0..x)?`** | Trigger voltage (Valt) | x=measuring range |
| **`TRIGGER EDGE (+/-)?`** | Edge + or – | |

Then, when the switch is set to **<P>**, the output of these parameters begins:

- Number of buffers (BUFNUM)
- Buffer length (BUFLENGTH) in bytes
- Trigger voltage (TRIGGER)
- Trigger edge (TR. EDGE)
- Scan time (SCANTIME)
- Table of all specified buffers with number, channel, and measurement range

| | |
|---|---|
| **`PRESS ENTER TO START SCAN`** | When the **ENTER** key is pressed, **`'SCANNING...'`** appears on the display and the system waits for triggering. Data acquisition can be aborted prematurely by briefly pressing the **BREAK** key. |
| **`SCREEN DISPLAY (Y/N)?`** | If Y is entered here, the extended **SCREEN** mode, whose commands were explained above, is entered. Here, too, the display **`'EXITWITH <E>'`** appears first. |
| **`PLOT (Y/N)?`** | Here, the output of the recorded signals to the plotter begins at **Y**. However, care must be taken to ensure that signals have not been captured by a repeated **INSCAN** with changed parameters. This will otherwise result in a program abort and an error message. |
| **`NEW SCAN (Y/N)?`** | New measurement data acquisition begins at **Y** |
| **`NEW PARAMETERS (Y/N)?`** | If **Y** is entered, the program restarts at the buffer number entry. If **N** is entered, the program terminates. |

# APPENDIX A

## Variable list of the Fourier modules

| | |
|---|---|
| A$ | Key code; magnitude symbol |
| P$ | Buffer for USING format; output string |
| Q$ | USING format |
| | |
| f | Flag for command transfer |
| I | Variable for counting loops |
| J | Intermediate values (e.g., when formatting a number) |
| K | Intermediate values |
| | |
| A0 | DC voltage component of a spectrum according to 'B' (volts) |
| AM | Maximum amplitude of the spectrum according to 'B' (volts) |
| AN | Amplitude of an individual value according to 'EW' (volts) |
| B1 | Buffer number of the acceleration signal (at 'D') |
| B1 | Buffer number of the 1st integral = velocity (at 'D') |
| B2 | Buffer number of the 2nd integral = distance (at 'D') |
| CB | Current buffer number (1..BUFNUM) |
| CP | Current cursor position (1..BUFLEN) |
| CS | Position on the screen (0..255) |
| CV | Compression factor (1/16..BUFNUM/256) |
| D0 | Factor for Buffer BO (for 'D' in m/s2/V) |
| D1 | Factor for Buffer B1 (for 'D' in m/s/V) |
| D2 | Factor for Buffer B2 (for 'D' in m/V) |
| DA | Position number of the lower limit (for 'D', 1..BUFLEN) |
| DE | Position number of the upper limit (for 'D', 1..BUFLEN) |
| DP | Flag for the output type (for 'D') |
| F1 | Base frequency of the FFT analysis according to 'B' (Hertz) |
| FL | Flag for spectrum (0: none, 1: linear, and 2: log. spectrum) |
| FR | Frequency of a single oscillation (for *EW in Hertz) |
| G1 | Factor for formatted output |
| G2 | Buffer for formatting factor for value output |
| GI | Grid step size for FFT spectrum labeling |
| IB | Buffer number (for 'T') |
| IC | Channel (for 'T') |
| IS | Scan time (for 'T') |
| IT | Trigger level (for 'T') |
| JF | Flag for offset type (for 'J') Integration factor (in seconds) |
| JO | Offset (for 'T' in value) |
| JP | Flag for output type (for 'S') |
| K0 | Number of signals to plot (1..4) |

| | |
|---|---|
| K0(5) | Contains scan time, history, and plot buffer |
| K2 | Quadratic harmonic distortion according to 'B' (0..1) |
| K3 | Cubic harmonic distortion according to' B' (0..1) |
| KL | Harmonic distortion according to 'B' (0..1) |
| MB | Marked buffer (M) |
| MP | Marked position (M) |
| NA | Number ' of periods for 'F' |
| NB | Buffer number for the integral for 'J' |
| NF | Nyquist frequency according to 'B' (Hertz) |
| NP | Marked position (C) |
| O0 | Offset for the acceleration signal (at 'D' in m/s2) |
| O1 | Offset for the velocity (at 'D' in m/s) |
| PA | Number of points for FFT analysis according to 'B' |
| PB | Buffer number for FFT analysis according to 'B' |
| PS | Number of the first point after 'B' |
| SC | Scan time in the current buffer (seconds) |
| SP | Current voltage value (volts) |
| ST | Step size for FFT spectrum labeling |
| TI | Current time (seconds) |
| VG | Number of points for the history |
| WA | First ordinal number for 'W' |
| WE | Last ordinal number for 'W' |
| WI | Angle of a single value after 'EW' |
| XX | ASCII value of the last key pressed (for heading) |
| ZB | Effective value of the ordinal range after 'R' (volts) |
| ZE | Effective value of the signal after 'B' (volts) |

# APPENDIX B

## Logical connection of the various commands

The commands of the extended **SCREEN** mode can, in some cases, only be called in a specific order. The diagram below illustrates the necessary command sequence. **J** is always started first. For example, the **<W>** key is only permitted if the area was first marked with **M** and **C** and the spectrum was formed with **B**. However, since an area has already been marked, other calculations and outputs for this area can also be performed with **P**, **J**, and **D**. The spectrum is retained until either buffer 1 is overwritten by **I** (**INSCAN**) or a new analysis is performed by **B**. The normal **SCREEN** mode commands, **M** and **C**, **S** and **F**, are always permitted, regardless of the marked range and the spectrum. The **G** command should only be used immediately after a double integration.